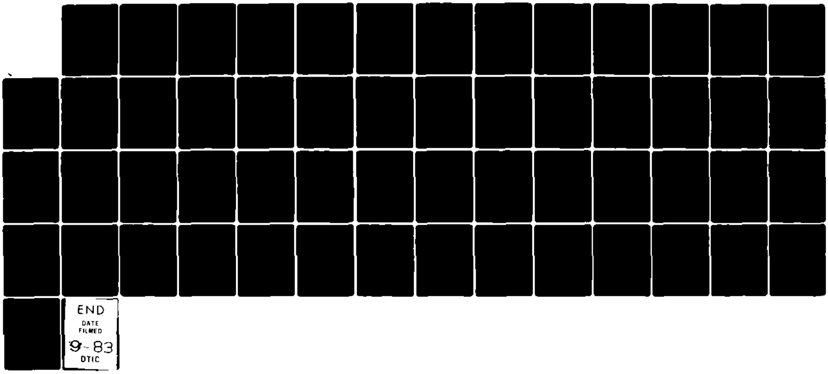
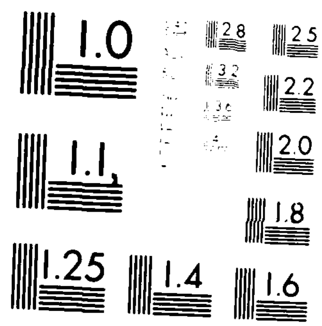
AD-A131 845 GSE FOR BALLOON-BORNE IMS: DECOMMUTATOR AND D/A UNITS 1/1
..(U) NORTHEASTERN UNIV BOSTON MASS ELECTRONICS
RESEARCH LAB R SUKYS ET AL. OCT 82 SCIENTIFIC-1
UNCLASSIFIED AFGL-TR-83-0095 F19628-81-C-0162 F/G 9/6 NL





MICROCOPY RESOLUTION TEST CHART

AFGL-TR-83-0095

GSE FOR BALLOON-BORNE I.M.S.: DECOMMUTATOR AND D/A UNITS

Raimundas Sukys
J. Spencer Rochefort

Northeastern University
Electronics Research Laboratory
Boston, Massachusetts 02115

Approved for public release; distribution unlimited.

Scientific Report No. 1

October 1982

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

This report has been reviewed by the ESD Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

This technical report has been reviewed and is approved for publication

Alan D. Bailey
Contract Manager

Rocco S. Narcisi
Branch Chief

FOR THE COMMANDER

C.G. Stergis
Division Director
Aeronomy Division

Qualified requestors may obtain additional copies from the Defense Technical Information Center. All others should apply to the National Technical Information Service.

If your address has changed, or if you wish to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify AFGL/DAA, Hanscom AFB, MA 01731. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document requires that it be returned.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFGL-TR-83-0095	2. GOVT ACCESSION NO	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) GSE FOR BALLOON-BORNE I.M.S.: DECOMMUTATOR AND D/A UNITS		5. TYPE OF REPORT & PERIOD COVERED Scientific Report No. 1
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) R. Sukys J.S. Rochefort		8. CONTRACT OR GRANT NUMBER(s) F19628-81-C-0162
9. PERFORMING ORGANIZATION NAME AND ADDRESS Northeastern University Electronics Research Laboratory Boston, MA 02115		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2310G3AW
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Geophysics Laboratory Hanscom AFB, Massachusetts 01731 Monitor/A.D. Bailey/LKD		12. REPORT DATE October 1982
		13. NUMBER OF PAGES 50
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Tech., Other

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Decommutator
Digital to Analog Converter

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Two ground based units to support data processing and control operations during the flight of a Balloon-Borne Ion Bass Spectrometer (BBIMS) are described. One of the units decommutated the incoming PCM data and presented that data upon request to a computer for further processing. The data was available through IEEE-488 compatible interface and/or in parallel form as 16

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

and/or 8 bit words. Communications interface for commands and messages to and from the airborne instrumentation was also provided. The second unit converted the digital data, supplied by the computer, into analog signals for display and recording. An IEEE-488 compatible interface was used for data transfer.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS

TABLE OF ILLUSTRATIONS

TABLE OF SELECTED ACRONYMS AND ABBREVIATIONS

1. ATN. . . . . ATTENTION
2. CMOS . . . . COMPLIMENTARY METAL OXIDE SEMICONDUCTOR
3. CPU. . . . . CENTRAL PROCESSING UNIT
4. CRT. . . . . CATHODE RAY TUBE
5. CS . . . . . CHIP SELECT
6. D/A. . . . . DIGITAL TO ANALOG
7. DAV. . . . . DATA VALID
8. D to A . . . DIGITAL TO ANALOG
9. EOI. . . . . END OR IDENTIFY
10. EPROM. . . . ELECTRICALLY PROGRAMMABLE READ ONLY MEMORY
11. FIFO . . . . FIRST IN, FIRST OUT
12. HEXFET . . . POWER METAL OXIDE SEMICONDUCTOR (Trade name of International Rectifier)
13. IFC. . . . . INTERFACE CLEAR
14. LED. . . . . LIGHT EMITTING DIODE
15. LSB. . . . . LEAST SIGNIFICANT BIT
16. μP . . . . . MICROPROCESSOR
17. NDAC . . . . NOT DATA ACCEPTED
18. NRFD . . . . NOT READY FOR DATA
19. PCM. . . . . PULSE-CODE MODULATION
20. RAM. . . . . RANDOM ACCESS MEMORY
21. RTS. . . . . REQUEST TO SEND DATA
22. SRQ. . . . . SERVICE REQUEST
23. TTL. . . . . TRANSISTOR-TRANSISTOR LOGIC
24. USART. . . . UNIVERSAL SYNCHRONOUS/ASYNCHRONOUS RECEIVER/ TRANSMITTER

INTRODUCTION

Two ground based units were developed to support a Hewlett Packard HP9845C system in data processing operations during the flight of a Balloon Borne Ion Mass Spectrometer (BBIMS). One of the support units was designated to decommutate the incoming PCM data, while the other was developed to convert digital data, supplied by the computer, into analog signals for display.

The data gathering modes of the mass spectrometer could be controlled from the ground. The instrument could be commanded to execute any one of a number of programs or entire repertoires of programs stored on board. Also, entirely new programs could be transmitted from the ground for execution over a serial command link available for this purpose. Commands and instructions are sent to the mass spectrometer control unit through a command transmitter. A single 8 bit word within the PCM data train was used as the down link for communications from the airborne unit. Detailed description of the BBIMS instrumentation and its capabilities can be found in References 1, 2 and 3. The flexibility of the airborne instrument allowed the ground based scientist to tailor the data gathering process to the existing conditions. This required an almost real time processing and display of data, and the capability to send commands to and to receive communications from the airborne mass spectrometer control unit. Also, a large number of signals were displayed to monitor the status and the performance of the balloon borne instrumentation.

During the first successful flight of the BBIMS instrumentation package in May 1982, the HP9845C system was used to display the monitor

signals and to plot selected portions of the ion spectrum. It received the data from a ground control unit designed primarily to communicate with the airborne control unit. The data was available in a parallel form, one byte at a time, together with a word synchronization pulse. Frame synchronization was not available; therefore, the computer had, for all practical purposes, to decommutate the data. This task consumed a considerable amount of processing time Commands to the mass spectrometer had to be sent through the ground control unit.

The newly developed system overcomes many of the disadvantages of the earlier ones and is shown in the block diagram of Figure 1.

The decommutator, one of the newly constructed ground support units, relieved the HP9845C system from the task of the PCM data decommutation. It presented the selected data, at the convenience of the computer, through the IEEE-488 bus.[4] Also, it provided a communications interface between the computer and the serial command link and relayed the messages from the balloon-borne mass spectrometer control unit, within the PCM data stream, to the HP computer. Provisions were included for the use of a printer to obtain a hard copy of all communications.

The D/A UNIT was designed to convert eight bytes of data into analog signals. Microammeters and output terminals were provided for the display and for the recording of the analog data. The data channels were selected by the computer.

This report describes the capabilities and the operation of these two support units. Detailed descriptions of the circuits are provided.

I. DECOMMUTATOR

The primary task of the BBIMS-2 Decommutator Unit was, as its name implies, to decommutate the incoming PCM data from the balloon borne instruments. The decommutated data was made available to the user through an IEEE-488 interface bus and also in a parallel form as 16 and/or 8 bit words. The parallel data was accompanied by appropriate synchronization pulses. The secondary task of the unit was to provide communications/command interface to the airborne mass spectrometer control unit. All communications passing through the interface were available at yet another port for use by a printer or a CRT terminal. The communications ports were compatible with the RS-232-C[5] link. LED indicators were provided to monitor the operation of the decommutator unit.

A. OPERATION

The incoming PCM data from the balloon borne instruments was formated into a 20 word frame transmitted at 12 kilobits per second. Eight bit words were used. A 16 bit pattern (EB90H) provided the frame synchronization. The subframe consisted of 96 data words. Two of the subframe words appeared in each frame. Those words were preceded by a subframe identification code. Most of the other words within the frame carried data associated with the quadrupole ion mass spectrometer, while the subframe was mostly used for monitor data or for the data from supporting experiments. A notable exception were two words within the frame. One contained the ONE's count of the frame, but did not include the frame synchronization pattern nor the ONE's in the count word itself. The count was intended to

show the number of non-canceling errors within the frame and, therefore, provided a rough indication of the quality of the telemetry link. The other word carried communications from the mass spectrometer flight control unit to the ground station. The communications words were inserted into the PCM data stream at an equivalent rate of 300 baud. This rate was chosen to accommodate the relatively slow ground support equipment. Also, since the data gathering rate of the quadrupole ion mass spectrometer was quite low, not every frame of the PCM pulse stream carried new and/or valid data. The first word of a frame was used to indicate the status of the data and the presence or absence of a new communications word within that frame. The decommutation process was controlled,to a large extent, by the status of two bits within that word.

Whenever the designated bit (MSB) in the first frame word indicated that a new and valid data was present in the frame, the data was extracted and stored. Only selected words were thus processed. The selection was determined by software. Primarily, the words directly associated with the data gathered by the ion mass spectrometer were chosen.

Eight 32 word registers were created for storage of that data in the 1K RAM included in the decommutator circuits. Each of the registers was dedicated to one minor frame. Thus eight minor frames of data could be stored consecutively before the very first frame of data was lost due to an overwrite. Although only 13 words needed storage locations, the registers were purposely extended to 32 locations. The balloon borne PCM encoder format was software programmable; therefore, future flights can utilize a different format if many more

words within a frame require storage in the RAM. To maintain this flexibility the very first location of each storage register was reserved for the number of words actually stored; therefore, only the locations used for storage were addressed during the transfer of the data to the peripheral processing equipment. Once the data was read, ZEROS were placed into the first location of the storage register indicating that the register has been processed.

Subframe data, which contained mostly the monitor signals of the airborne instrumentation was extracted and stored continuously. For that purpose two 256 byte registers were allocated in the RAM. Once again, the registers were able to accommodate considerably more data than was available in each subframe; therefore, future subframes may contain more data. The subframe data words were stored consecutively. Once all of the subframe words were stored, a binary number representing the number of words contained in the register was placed into the first location. During the following major frame of the PCM data the second storage register was filled. By that time, hopefully, the data from the first storage register were transferred to the peripheral sources. If not, then an overwrite occurred. The decommutation process was controlled by the subframe identification word which had to precede the first subframe word within a minor frame. A subframe identification word 00H was interpreted as a beginning of the subframe.

The data stored in the RAM was available only through the IEEE-488 interface. A subroutine searched for a full storage register containing the mass spectrometer data. Once found, the decommutator established contact with the bus controller and then transmitted the

stored data. When transfer was completed, the program returned to check if the next consecutive register containing the mass spectrometer data was full. Thus the data transmission was carried out on FIFO basis. Only when all eight of the frame storage data registers were empty, the contents of a full subframe data storage register were output. Therefore, the mass spectrometer data had a priority over the subframe data. To differentiate between the two data types, an identification code, six for the subframe data, seven for the minor frame data, was transmitted as the first word. The word count stored in the first location of the storage register was not transmitted. The end of the data transfer was marked by a TRUE state on the EOI line during the transmission of the last data word. Provisions were included to continue data transmission from the same storage register following an interruption by IFC command from the controller. When the frame synchronization was lost the data processing was suspended.

The parallel data was available in real time. It remained stable for the duration of approximately one PCM data word. The 16 bit words were presented with the odd numbered PCM data word as the most significant byte. A word sync pulse lasting one-half of the duration of a PCM word and centered in that time period marked the availability of the 16 bit word. The byte wide data was also marked with a similar pulse on a different output line. Finally, when the whole synchronization word (16 bits) was available at the parallel data port, a synchronization pulse coinciding with and of the same duration as the second frame synchronization byte (90H) appeared on the third output line. The parallel data, although its integrity could be questionable, was available even when the frame synchronization was lost.

The communications from the airborne mass spectrometer control unit to the ground station were processed in somewhat similar manner as the PCM data. When a selected bit (3rd. USB) in the first word of the PCM data train indicated that a new communications word was present in the frame, that word was extracted and stored. A 256 byte register was allocated in the RAM for that purpose. Next, the status of the decommutator USART, used to retransmit the communications from the flight control unit to the xontroller (HP9845C), was checked. When it was found to be empty, the previously stored word or, it none were available, the very recent word was transferred into the USART. When no new communications were present in the frame, then only the USART was serviced during tne time interval allocated for the communications word within the frame.

B. PROGRAM

The program controlling the operations of the decommutator unit could be subdivided into three groups of subroutines each performing distinct tasks. The system was prepared for operation by the initialization routines. Data transfer and handshakes with the bus controller were handled by another set of subroutines. The decommutation and storage process was controlled by a subprogram designed to accommodate the frame format. This latter program called upon appropriate subroutines to perform the necessary decommutation and storage tasks. The decommutation subprogram was entered from the data transfer subroutines upon an interrupt generated by the hardware. Most of the time the system operated in the data transfer group of subroutines.

Upon TURN-ON or RESET the program initialized the decommutator ports, established clock frequency and continued into a subroutine

which determined the polarity of the incoming PCM data. Signal conditioning circuits were set to present the data in the proper polarity to the decommutator shift registers. Initial frame synchronization was also established at this point. From there, the system entered a routine where a search was conducted for a full decommutated data storage register. When a full register was found, another routine requested service from the controller of the IEEE-488 bus. After the designation to be a TALKER (code 29D) has been received, another subroutine transferred data from the storage register to the bus. When the transfer of the data from that register has been completed the system returned to the routine which searched for another full storage register.

The tasks of the data transfer from the storage registers in the RAM to the bus were interrupted whenever a PCM data word was ready for processing. Upon interruption, the contents of all CPU registers were saved. The control was transferred to the decommutator subprogram which selected an appropriate subroutine to process that particular word within the PCM data frame. To adapt the decommutator to a different PCM data format only the decommutator subprogram had to be changed. The subprogram consisted mainly of a series of jump instructions transferring control to the various processing routines. Upon completion of a given task, the program restored the contents of the CPU registers and returned to continue the interrupted data transfer process. The subroutines of the program and their flow graphs are presented in the APPENDIX at the end of this report.

C. CIRCUITS

The decommutator design was based on the 8085 µP and its family of support components. Some of the support functions were delegated to a sequential logic utilizing CMOS components. The circuit diagram is shown in Figure 2.

The incoming PCM data was transmitted through U20, U25 and the AND-AND-OR gate arrangement (U27) to a 16 bit shift register (U10, U11). These gates were used to invert the PCM data when necessary. Once the program determined, during the initialization process, that an inversion of polarity was required, a negative level was applied to pin 2 of U27 through the port C of U4. This process was non-reversible unless the system was reinitialized.

The clock to enter the data into the serial-in-parallel-out shift register (U15, U16) was derived from the CPU clock through U4 and U24. Synchronization with the incoming PCM data was maintained by resetting U24 through the AND-AND-OR gates (U26) on every transition in the data bit stream (transitions marked by pulses generated by U25) or by utilizing the external clock supplied with the PCM data. When present, the external clock was delayed one quarter of a period by U18 and its period was doubled by U19. After passage through U26 the transitions of this signal caused U25 to generate pulses which were then used to synchronize the internal clock with the incoming data.

The switchover from the data controlled synchronization to the external clock synchronization was accomplished in the gates of U26 in conjunction with the signals generated by U18. This monostable

C. CIRCUITS

The decommutator design was based on the 8085 ..P and its family of support components. Some of the support functions were delegated to a sequential logic utilizing CMOS components. The circuit diagram is shown in Figure 2.

The incoming PCM data was transmitted through U20, U25 and the AND-AND-OR gate arrangement (U27) to a 16 bit shift register (U10, U11). These gates were used to invert the PCM data when necessary. Once the program determined, during the initialization process, that an inversion of polarity was required, a negative level was applied to pin 2 of U27 through the port C of U4. This process was non-reversible unless the system was reinitialized.

The clock to enter the data into the serial-in-parallel-out shift register (U15, U16) was derived from the CPU clock through U4 and U24. Synchronization with the incoming PCM data was maintained by resetting U24 through the AND-AND-OR gates (U26) on every transition in the data bit stream (transitions marked by pulses generated by U25) or by utilizing the external clock supplied with the PCM data. When present, the external clock was delayed one quarter of a period by U18 and its period was doubled by U19. After passage through U26 the transitions of this signal caused U25 to generate pulses which were then used to synchronize the internal clock with the incoming data.

The switchover from the data controlled synchronization to the external clock synchronization was accomplished in the gates of U26 in conjunction with the signals generated by U18. This monostable

detected the presence of the external clock, generated the switchover signals and lit a LED to indicate to the user that indeed the external clock was present.

The frequency of the data shift clock output of U24 was divided by 8 in the bit counter of U28. The output of the counter was used to interrupt the μP and to latch the data present in the shift register into the parallel data output ports U15 and U16. The synchronization pulses for the user of the parallel data were derived in units 29 and 30 from the signals of U28.

The overall synchronization between the PCM data and the sequential logic circuits was maintained by software. At the beginning of the subroutine which processed the second byte of the frame synchronization word (90H) a pulse from the LSB of port C of U4 reset the bit counter U28. From there on, provided the data and/or the external clock did not have noise induced transitions, the system was synchronized. When the synchronization was lost, it was readily restored when the CPU again detected the PCM frame synchronization word. During the time that the CPU searched for the frame synchronization pattern, additional reset pulses were generated whenever the byte EBH was detected. These pulses were necessary to resynchronize the bit counter in order that the second byte (90H) of the synchronization word could be detected eight bits later. Therefore, the BYTE, the WORD and the FRAME synchronization pulses for the parallel data output were not valid during the search periods.

A LED was used to signal the synchronization status. In absence of signal the light was off. Random signals with an occasional frame synchronization byte pattern produced a flickering light. Synchronization was marked by a steady glow.

Upon receiving an interrupt from the bit counter (U28), the CPU transferred the data byte from the shift register U11 through port A of U5 into the scratch pad RAM of U4. From there, only selected data words were transferred into one of the storage registers in the 1k byte RAM U14.

The communications from the balloon borne ion mass spectrometer control unit to the ground control were transmitted through the USART U17 and the SR-232C interface circuits U21 and U22. When ready to transmit, the USART generated a Request To Send signal ($\overline{RTS}$). When the request was granted ($\overline{CTS}$ low), transmission of the messages started (via P1-2) and continued until all of the stored messages were transmitted. The transmission could be interrupted by canceling the permission to send ($\overline{CTS}$ high). The USART resumed transmission of the interrupted message when the $\overline{CTS}$ signal was once again received. The end of transmission was indicated when the USART removed the $\overline{RTS}$ signal from the line. Although the software to receive data through the USART from the command link receiver was not included in the program, hardware connections to the USART were provided for that purpose.

Commands from the ground based control to the mass spectrometer flight control unit arriving on P1-2, were transmitted to the command transmitter via S2-14. The decommutator unit acted as a junction box between the two units and provided an isolated connection to the printer port. The output of the USART was OR'ed with the command signals before entering the port.

The IEEE-488 bus transactions were carried out through U4 and U5. U4 received the signals from the bus which U5 transmitted to

the bus. Port A of U4 monitored the data bus. Port B of the same unit handled the handshakes and the commands. Data transmission took place through port B of U5 in conjunction with the buffers U13. The handshakes were transmitted through port C of U5 and the buffers U12.

Port C of U4 was designated as a general utility port. The synchronization pulse for the bit counter and the signal to control the polarity of the incoming PCM data were transmitted through this port. Also, LED indicators to display the status of the decommutator unit were controlled through that port.

The DAV LED was turned on when the data within the PCM data frame was new and valid. The TTY LED was lit when a communications word was present in the frame. The SR LED indicated that a service request (SRQ) was issued to the controller by the decommutator unit. Finally, the TALK LED turned on when the decommutator started data transmission over the IEEE-488 bus. These LED's were under software control and were turned on for a minimum of 3 PCM data frames.

The chip enable functions were generated by the decoder U2, while U9 latched the low byte of the address for the EPROM U8, where the decommutator program was stored.

II. D/A UNIT

To observe the data and/or the monitor signals received from the balloon-borne instrumentation an eight channel digital to analog converter unit was constructed. It was designed to be compatible with the IEEE-488 bus.

Its task was to receive an eight bit digital data word from the bus controller, to perform digital to analog conversion and to display the resulting analog signal on a designated microammeter. Each of the analog signals was also available on two output connectors for recording devices.

The D to A converter unit was designed to be a listener only. It could be addressed with the decimal codes 22 or 30 followed by one of the secondary codes 16 through 23. The secondary codes designated one of the eight D to A converters to be used for the conversion and the display of the digital data word.

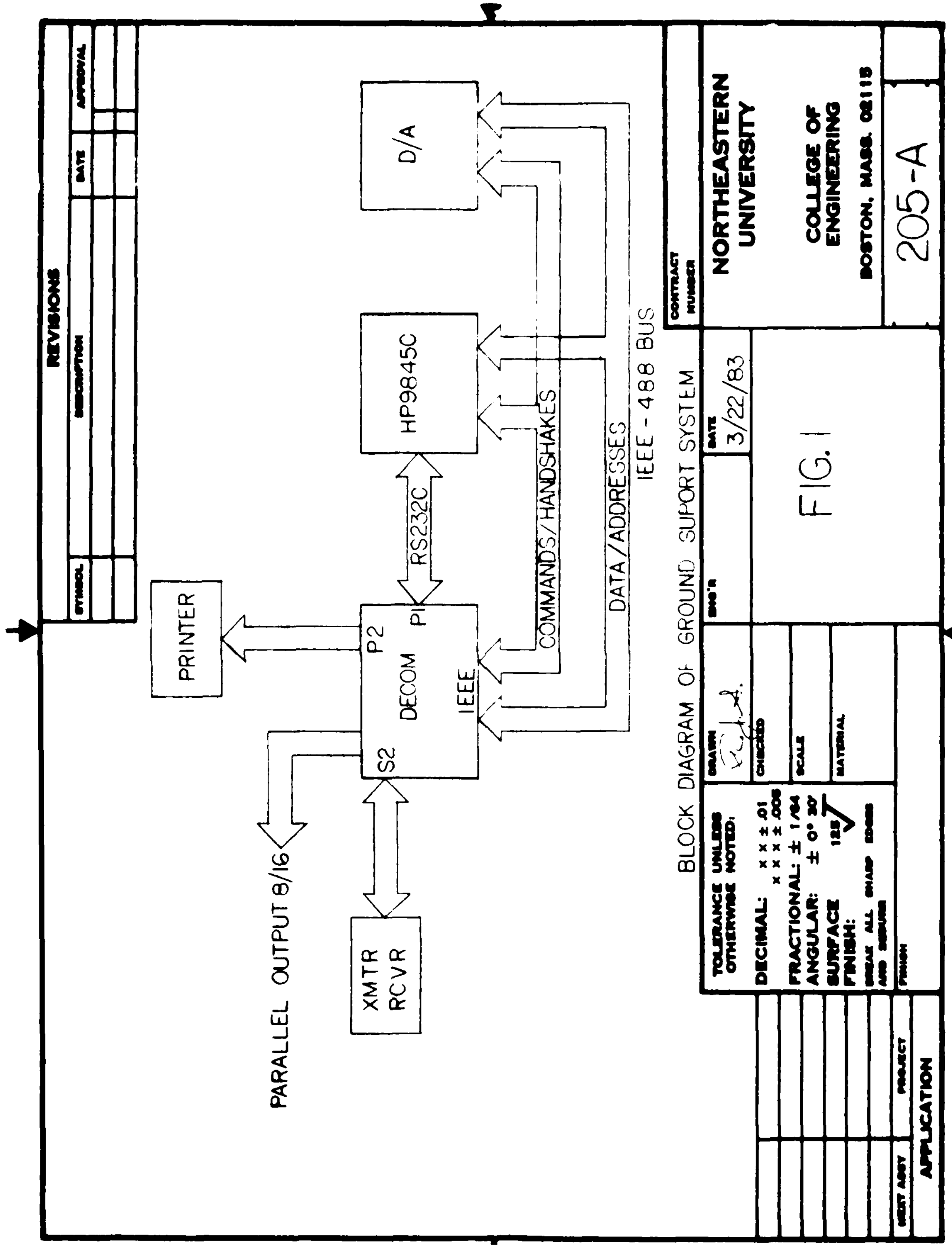
The circuit diagram for the D/A unit is shown in Figure 2. With the exception of the circuits directly interfacing with the IEEE-488 bus, standard CMOS devices were used to implement the necessary control logic. A TTL compatible high speed CMOS unit (U10) transmitted the data bus signals to the internal bus. Low power Schottky TTL gates (U1A,B) received the control signals and the low power HEXFETS (U2A,B) were used to perform the handshakes. To translate from the TTL levels of the IEEE-488 bus related devices to the CMOS levels, pull-up resistor network (U11) was used. The inverters (U3A,B) and the gates (U9) detected the presence of the primary select code on the internal bus. If at the same time the ATN line was in the TRUE state, the output of U9B enabled the gate U7A. A transition of the DAV line to the TRUE state produced a positive pulse at the output Q2 of the monostable U5A, which through the gate U7A SET the flip-flop U6A. Thus the primary select code was detected and the gates U8 were set to receive the secondary select code. When one of the eight secondary codes appeared on the internal bus, gates U8 became enabled. Next,

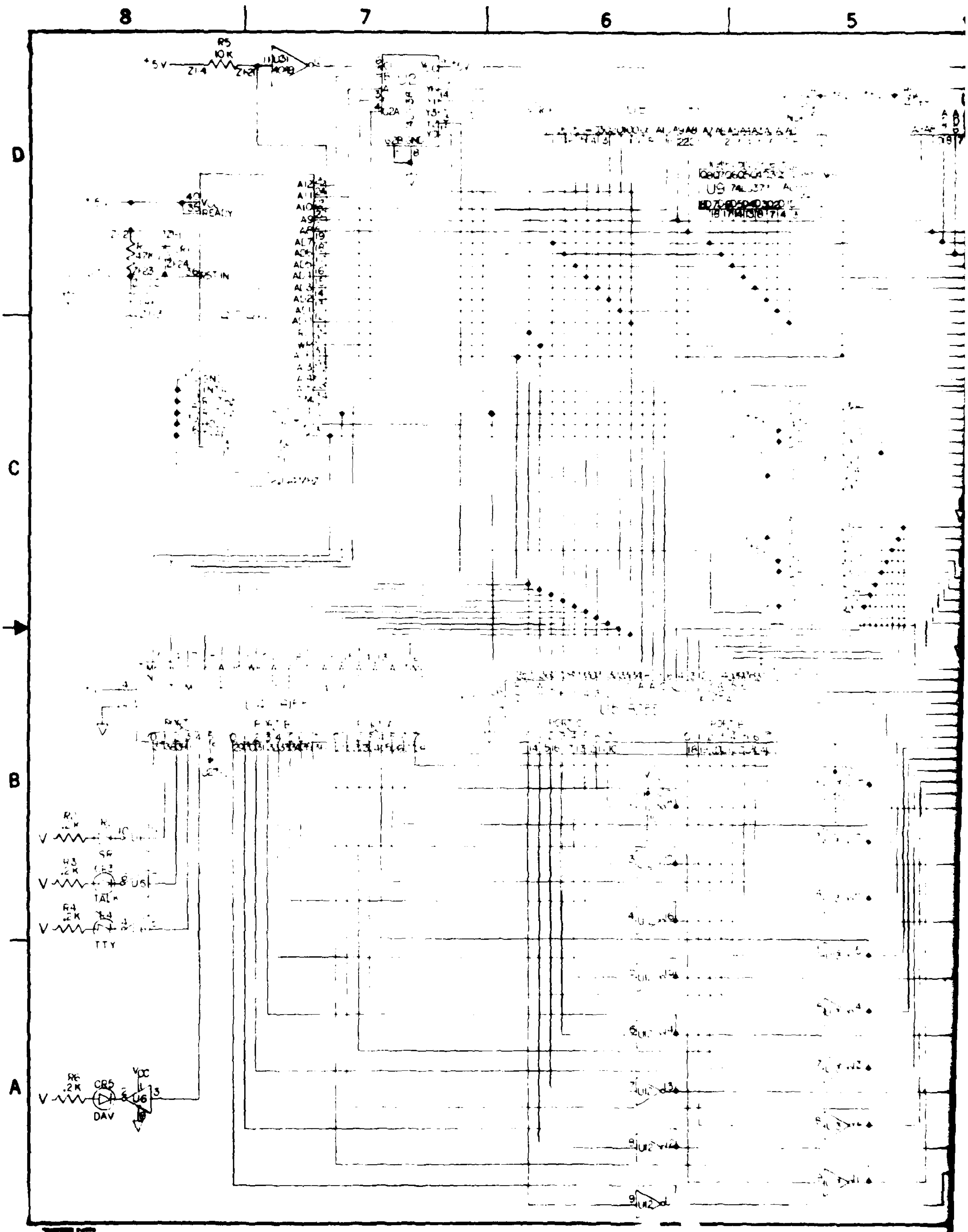
when the DAV line made the transition into the TRUE state, the four least significant bits of that code were latched into U12 by a pulse from U5A passing through the gate U8B. At the same time U4A was SET. Output Q_2 of U4A enabled the gate U7B. When the ATN line returned to the FALSE state, signifying that the next byte on the bus was data, the output of U12 was enabled. The ATN signal was passed through U1A, U3C, U7B and U3E. Thus one of the D to A converters (U13-U20) was selected ($\overline{CS}$ low) by U12 responding to the previously latched secondary code. The digital data word present at that time on the internal bus was latched into the internal register of the selected D to A converter by a pulse originating from $\overline{Q}_2$ of U5A. Once again that pulse was generated by the transition of the DAV signal from FALSE into the TRUE state. Since latching occurred on the trailing edge of the pulse, setting time for the data and the set-up time for the converter was provided.

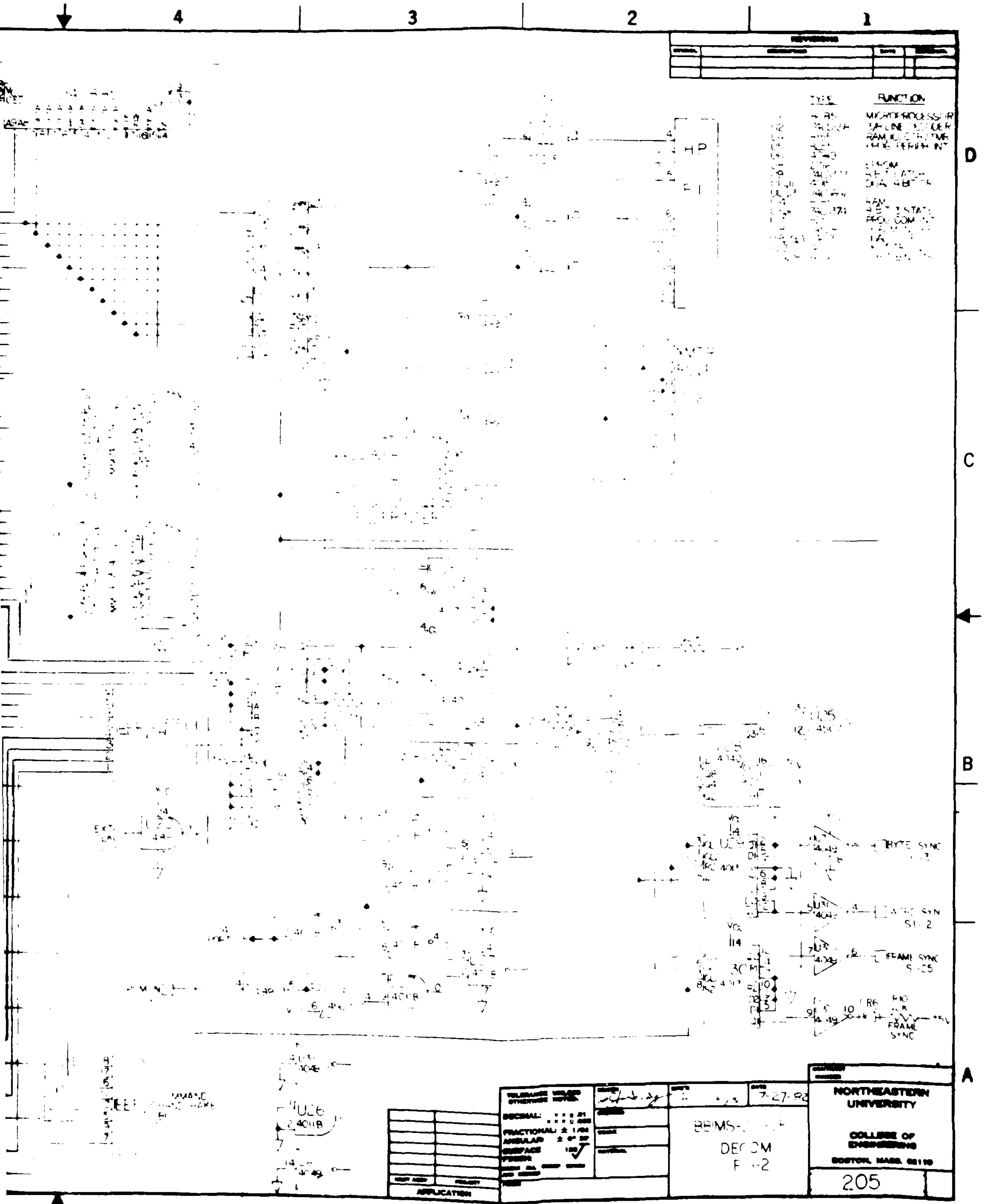
To indicate to the user of the unit, that data transfer into the D to A converter had taken place, the LED CR3 was lit whenever the outputs of the U12 were enabled. Flip-flop U6B generated the pulse to turn the LED on.

FALSE state on the ATN line disabled the code selection gates. Transitions of the DAV line from TRUE to the FALSE state with the ATN line in the FALSE state prepared the secondary code detection circuits for the next select code. The transitions generated pulses at U5B. These pulses passing through U7C clocked the flip-flops U6A and U4A into the reset states. The secondary selection code detection gates U8A became disabled. Also, the output of U12 returned to the ONE state. Thus the A to D converters were deselected.

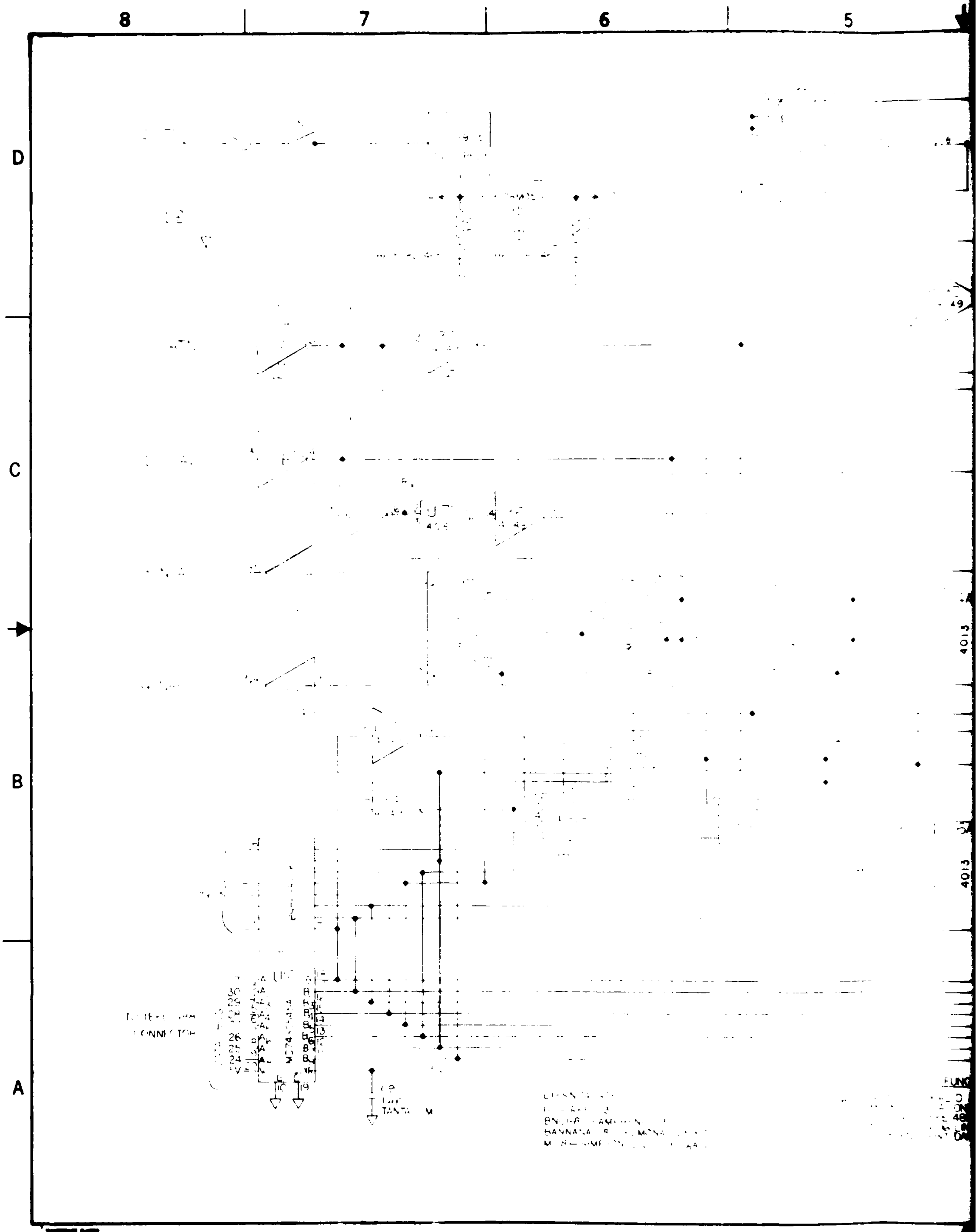
The handshake signals were generated by U4B. A pulse from $\overline{Q}_2$ of U5A produced the NRFD and $\overline{NDAC}$ outputs. A pulse from $\overline{Q}_1$ of U5B at the trailing edge of the DAV signal returned the handshake lines to $\overline{NRFD}$ and NDAC states. Initialization pulse for the control circuits at power turn-on was generated by U3D.

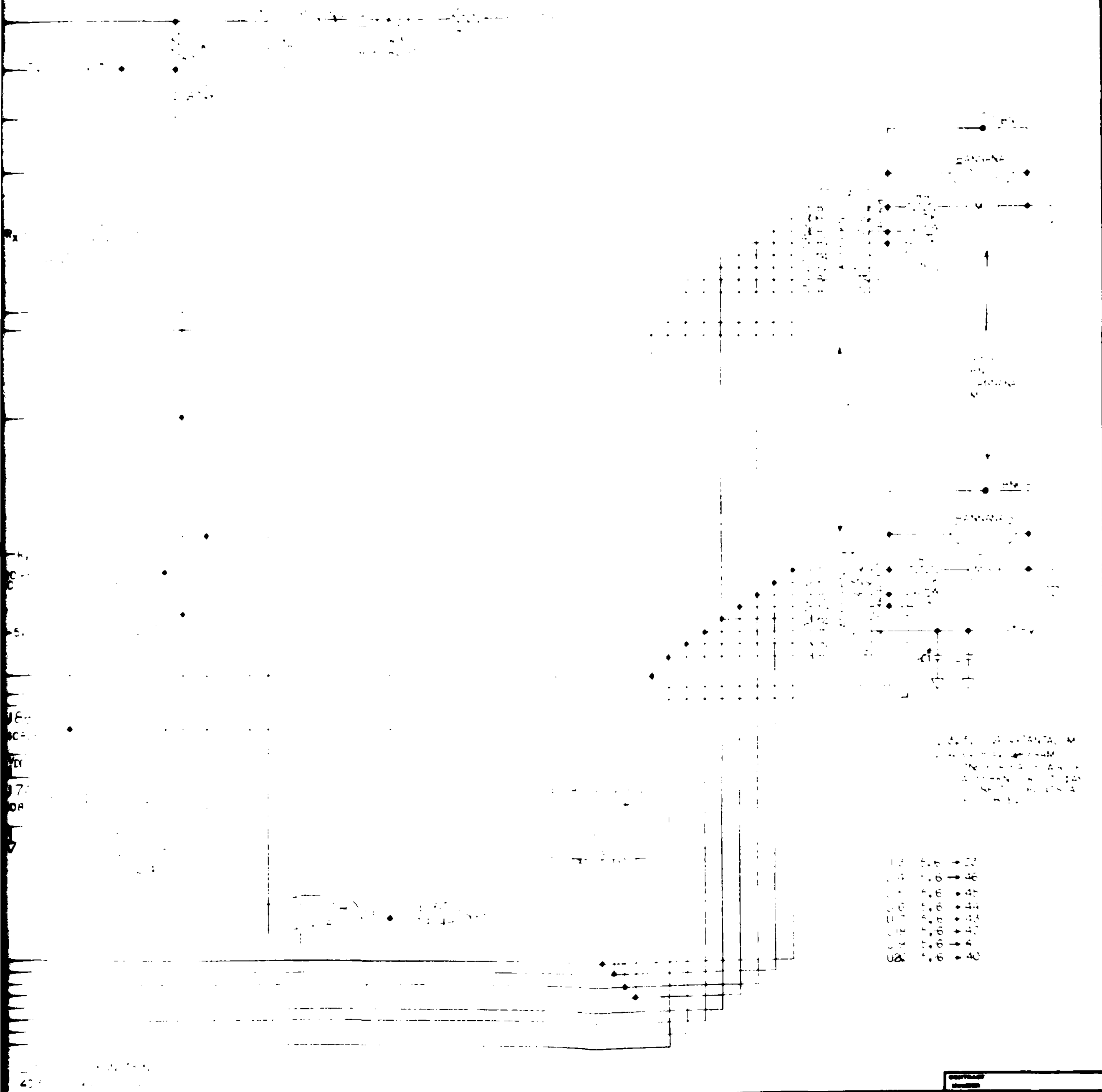

REVISIONS
SYMBOL
DESCRIPTION
DATE
APPROVAL
PRINTER
P2
P1
DECOM
IEEE
S2
RS232C
HP9845C
D/A
COMMANDS/HANDSHAKES
DATA/ADDRESSES
IEEE-488 BUS
PARALLEL OUTPUT 8/16
XMTR
RCVR
BLOCK DIAGRAM OF GROUND SUPORT SYSTEM
TOLERANCE UNLESS OTHERWISE NOTED:
DECIMAL: x x ± .01
x x x ± .005
FRACTIONAL: ± 1/64
ANGULAR: ± 0° 30'
SURFACE FINISH: 125
BREAK ALL SHARP EDGES AND DEBURR
FINISH
DRAWN
CHECKED
SCALE
MATERIAL
ENG'R
DATE
3/22/83
FIG. 1
CONTRACT NUMBER
NORTHEASTERN UNIVERSITY
COLLEGE OF ENGINEERING
BOSTON, MASS. 02115
205-A
NEXT ASSY
PROJECT
APPLICATION


8
7
6
5
D
C
B
A
R5
10K
+5V
U31
U9
R3
R4
TTY
R6
CR5
DAV
U6
VCC



4
3
2
1
D
C
B
A
REVISIONS
TYPE
FUNCTION
MICROPROCESSOR
H P
BYTE SYNC
FRAME SYNC
FRAME SYNC
APPLICATION
7-27-82
DECOM
NORTHEASTERN UNIVERSITY
COLLEGE OF ENGINEERING
BOSTON, MASS. 02115
205


8
7
6
5
D
C
B
A
4013
4013
CONNECTOR



REVISIONS

TOLERANCE UNLESS OTHERWISE NOTED

DECIMAL

FRACTIONAL: ± 1/64

ANGULAR

SURFACE FINISH

APPLICATION

BRIMS-2 SSE
D/A UNIT
FIG 3

NORTHEASTERN UNIVERSITY

COLLEGE OF ENGINEERING

BOSTON, MASS. 02115

211

APPENDIX - BBIMS-2 DECOM SUBROUTINES

TABLE OF CONTENTS

DECOMMUTATOR SUBROUTINES

XFR

This subroutine is entered upon the RST 7.5 interrupt. It transfers the present PCM data word into a temporary storage. The contents of all registers are saved in the stack. The appropriate address for the DCM routine is established and then transferred into the program counter.

DCM

This routine controls the decommutation by calling upon appropriate subroutines to process the words of the PCM data frame. To accommodate other PCM data frame formats the routine must be changed.

BK

The registers are reloaded from the stack. Interrupts are enabled and the system is returned to the interrupted task. This routine is entered at the end of every subroutine called by DCM.

WRD

When valid data is present in the PCM data frame, a selected word is stored in the buffer RAM. The storage locations within the buffer and the stored word counter are controlled by the routine.

WRD 1

This routine determines if new data and/or communications from the BBIMS flight control unit are present in the PCM data frame. It sets appropriate flags (DV and TTY) and lights the monitor LED's.

TTY

Communications from the BBIMS flight controller are stored into a temporary buffer within the RAM. When USART is empty a stored character is transferred into the USART for transmission to the computer.

SFID

When the subframe identification word is a binary ZERO, the contents of the subframe word counter are transferred into the first location of the subframe buffer. Pointers for the new storage buffer and the full buffer to be read out are established.

SFWD

This routine stores the subframe words and keeps count of the number of words stored.

FS1

Checks for the first frame sync word. If the word is not found it sets an error flag. When DV flag is detected the routine closes the present minor frame storage buffer by placing the stored word count into the first buffer location. New storage buffer address and pointer are established. In the case that the new storage buffer is in the process of being read out, the read pointer is pushed to the next buffer. Also the turn-offs of the monitor LED's are controlled by this routine.

FS2

Outputs synchronization pulse to the PCM data bit counter. Checks for the second frame sync word and re-establishes frame sync when necessary.

SEARCH

On turn-on or reset the search routine is entered as a part of an initialization process. It establishes the polarity of the incoming PCM data and the original frame sync for the decom system. From SEARCH the program enters IDLE.

IDLE

In this routine the system searches for full storage buffers. The minor frame data buffers have the priority. Only when all of these are empty the subframe data buffers are checked.

E488

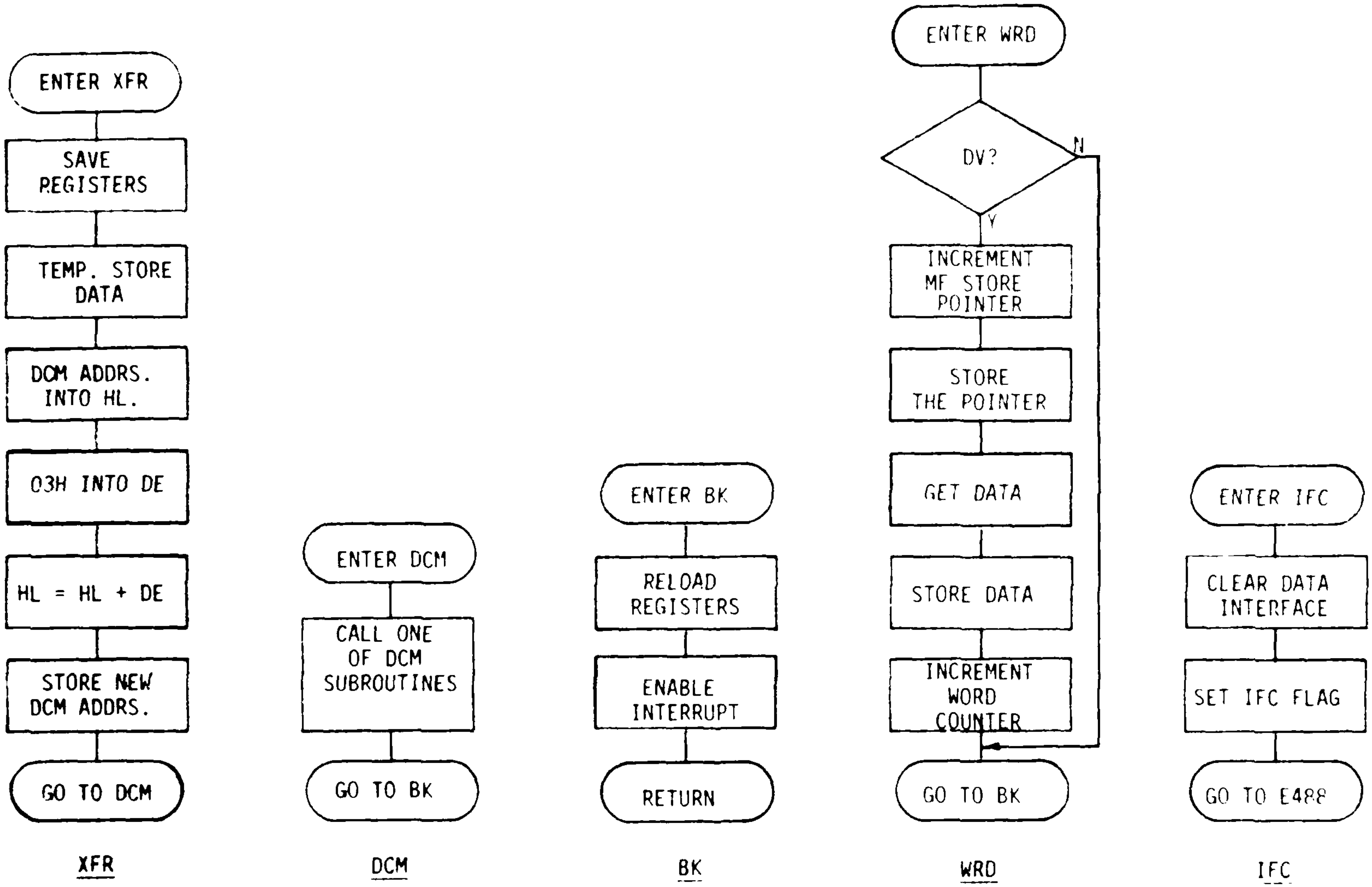
This routine requests service and establishes the first contact with the computer before data transmission. Parallel polling ID method is used.

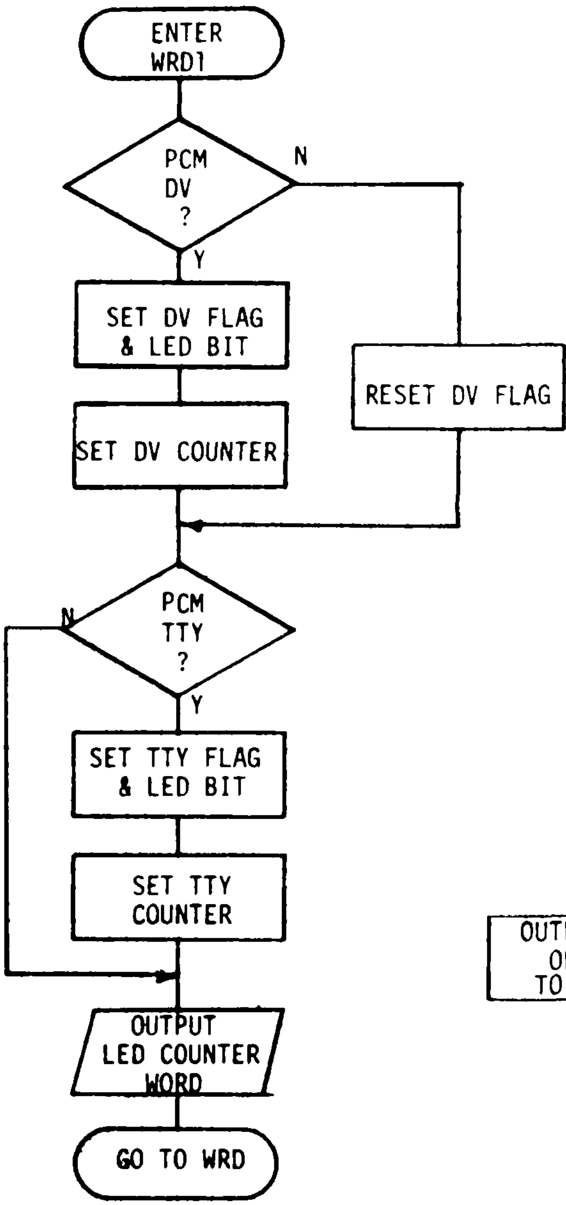
OUTPUT

Performs the necessary handshakes and transfers data from the storage buffers of the decom to the computer. The first data word identifies the data as coming from a subframe or a minor frame buffer. Data transfer ends with the EOI control code.

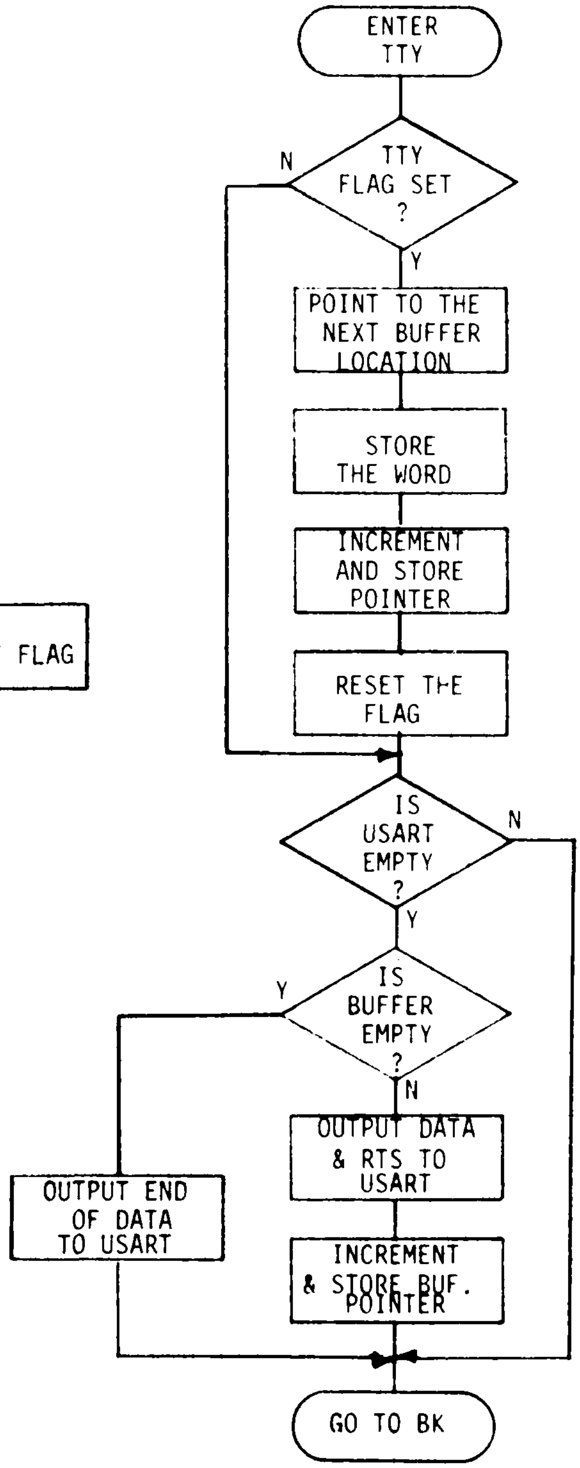
IFC

Clears the data interface upon detection of the IFC command during the data transmission. E488 is entered from this routine.

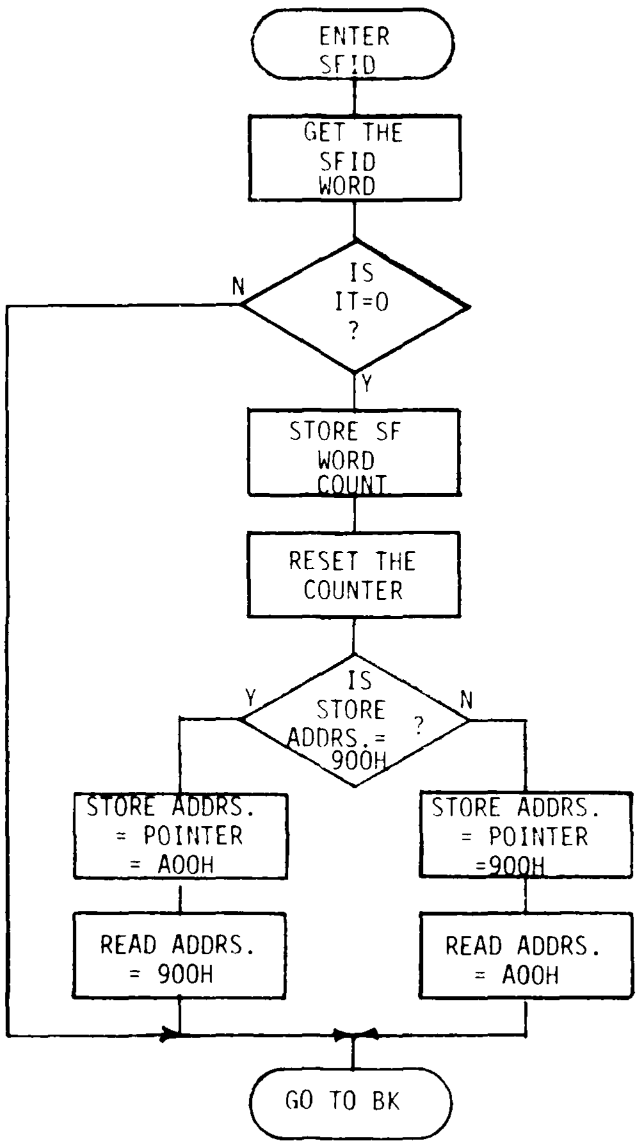

ENTER XFR
SAVE REGISTERS
TEMP. STORE DATA
DCM ADDRS. INTO HL.
03H INTO DE
HL = HL + DE
STORE NEW DCM ADDRS.
GO TO DCM
XFR
ENTER DCM
CALL ONE OF DCM SUBROUTINES
GO TO BK
DCM
ENTER BK
RELOAD REGISTERS
ENABLE INTERRUPT
RETURN
BK
ENTER WRD
DV?
N
Y
INCREMENT MF STORE POINTER
STORE THE POINTER
GET DATA
STORE DATA
INCREMENT WORD COUNTER
GO TO BK
WRD
ENTER IFC
CLEAR DATA INTERFACE
SET IFC FLAG
GO TO E488
IFC


ENTER WRD1
PCM DV ?
N
Y
SET DV FLAG & LED BIT
SET DV COUNTER
RESET DV FLAG
PCM TTY ?
N
Y
SET TTY FLAG & LED BIT
SET TTY COUNTER
OUTPUT LED COUNTER WORD
GO TO WRD

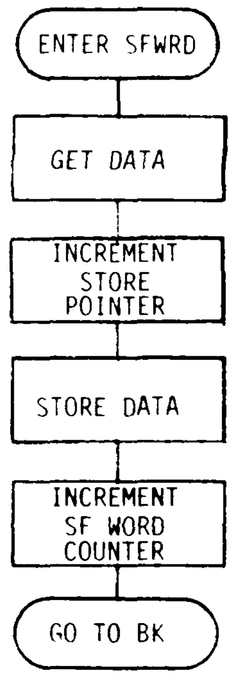
WRD 1


ENTER TTY
TTY FLAG SET ?
N
Y
POINT TO THE NEXT BUFFER LOCATION
STORE THE WORD
INCREMENT AND STORE POINTER
RESET THE FLAG
IS USART EMPTY ?
N
Y
IS BUFFER EMPTY ?
Y
N
OUTPUT DATA & RTS TO USART
INCREMENT & STORE BUF. POINTER
OUTPUT END OF DATA TO USART
GO TO BK

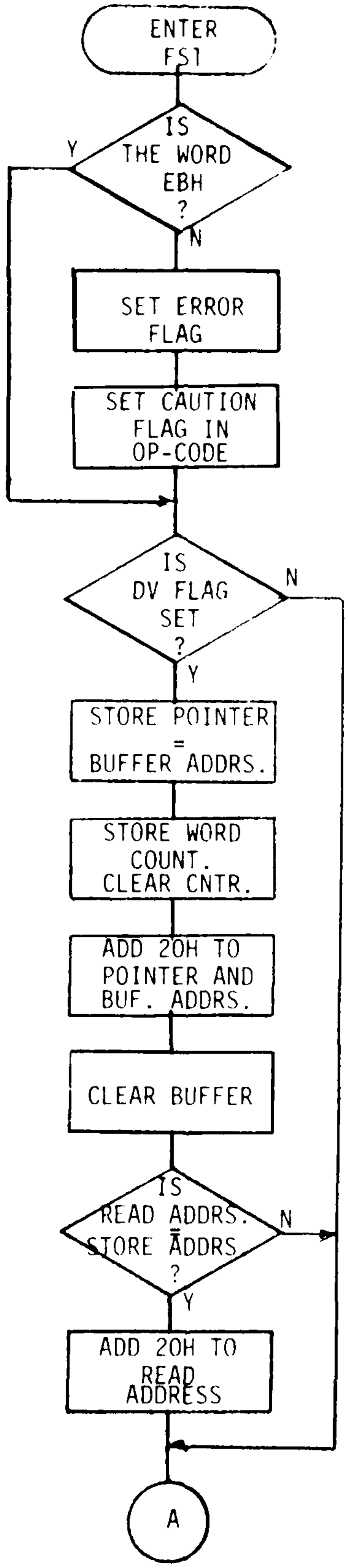
TTY


ENTER SFID
GET THE SFID WORD
IS IT=0 ?
N
Y
STORE SF WORD COUNT
RESET THE COUNTER
IS STORE ADDRS.= 900H ?
Y
N
STORE ADDRS. = POINTER = A00H
READ ADDRS. = 900H
STORE ADDRS. = POINTER =900H
READ ADDRS. = A00H
GO TO BK

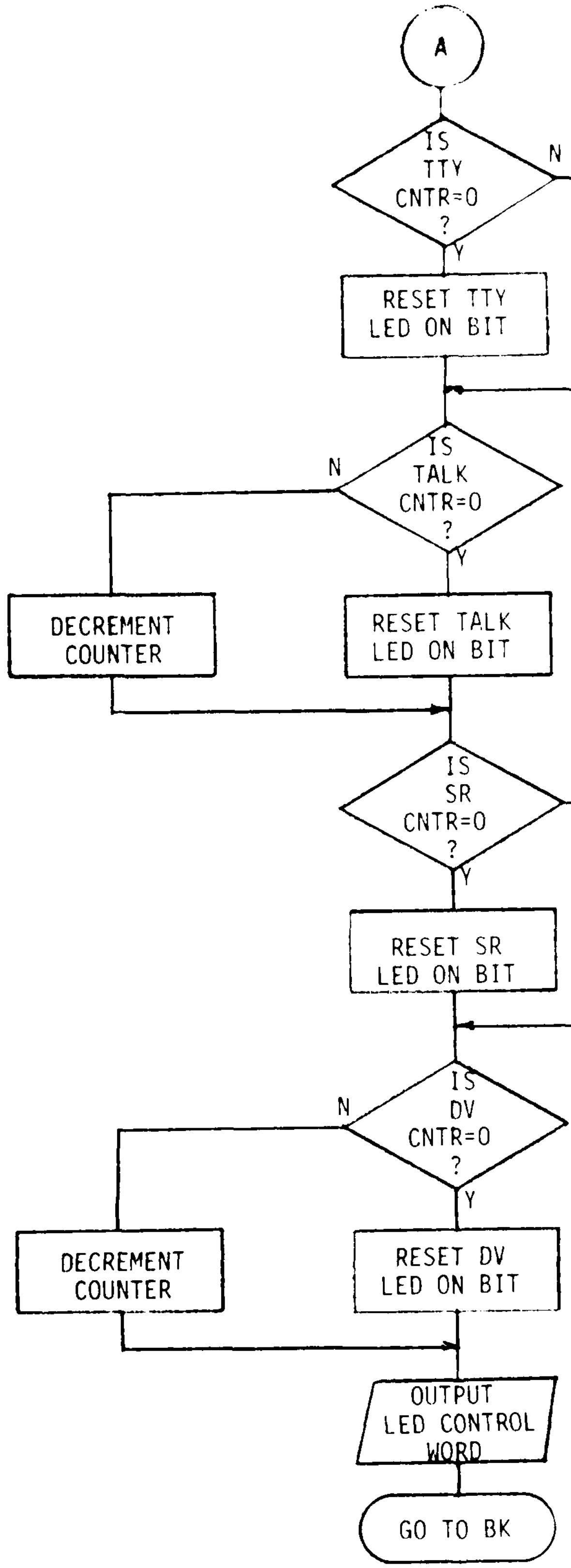
SFID


ENTER SFWRD
GET DATA
INCREMENT STORE POINTER
STORE DATA
INCREMENT SF WORD COUNTER
GO TO BK

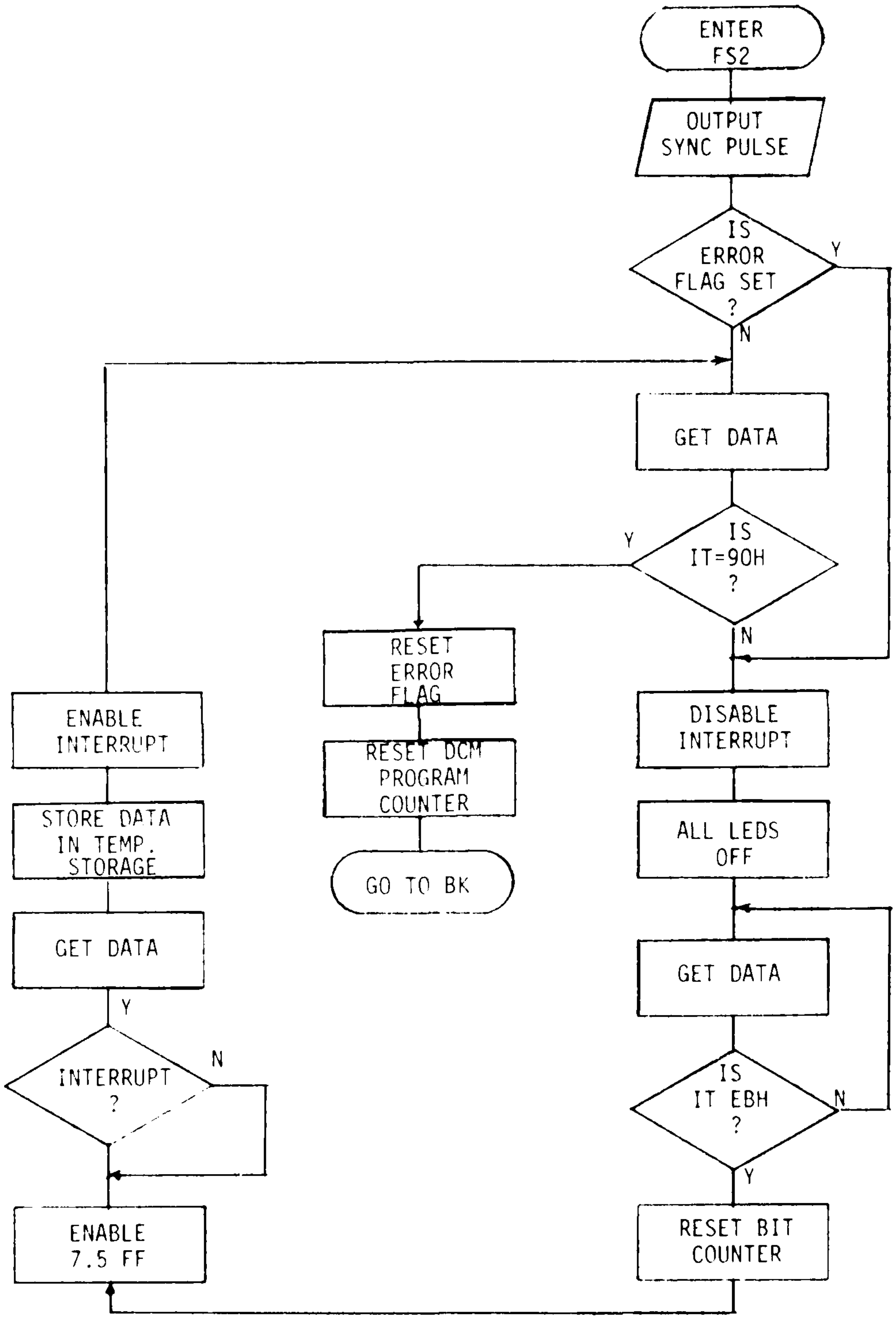
SFWRD


ENTER FS1
IS THE WORD EBH ?
Y
N
SET ERROR FLAG
SET CAUTION FLAG IN OP-CODE
IS DV FLAG SET ?
N
Y
STORE POINTER = BUFFER ADDRS.
STORE WORD COUNT. CLEAR CNTR.
ADD 20H TO POINTER AND BUF. ADDRS.
CLEAR BUFFER
IS READ ADDRS. STORE ADDRS ?
N
Y
ADD 20H TO READ ADDRESS
A

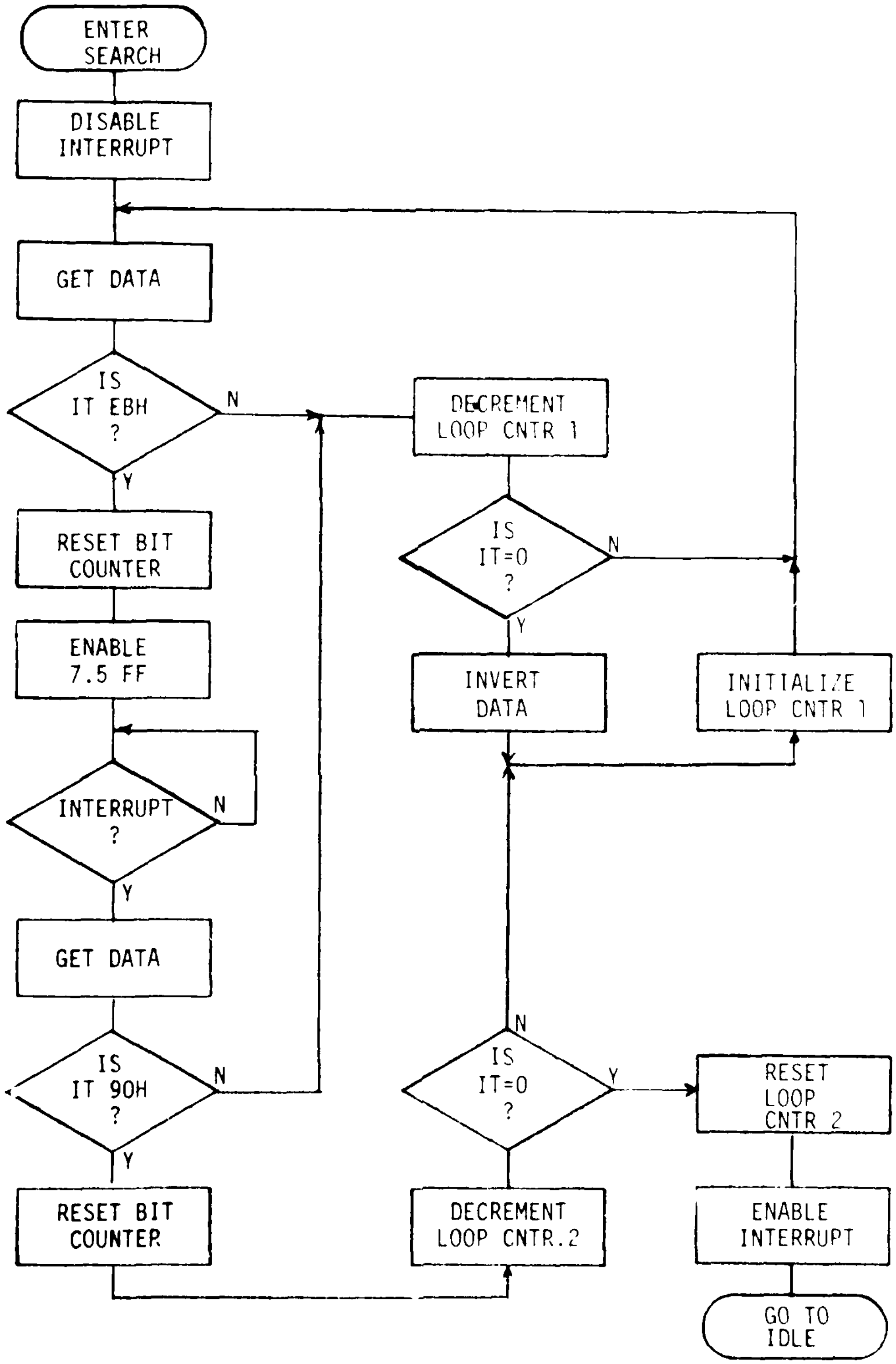
FS1


A
IS TTY CNTR=0 ?
N
Y
RESET TTY LED ON BIT
DECREMENT COUNTER
IS TALK CNTR=0 ?
N
Y
DECREMENT COUNTER
RESET TALK LED ON BIT
IS SR CNTR=0 ?
N
Y
RESET SR LED ON BIT
DECREMENT COUNTER
IS DV CNTR=0 ?
N
Y
DECREMENT COUNTER
RESET DV LED ON BIT
OUTPUT LED CONTROL WORD
GO TO BK

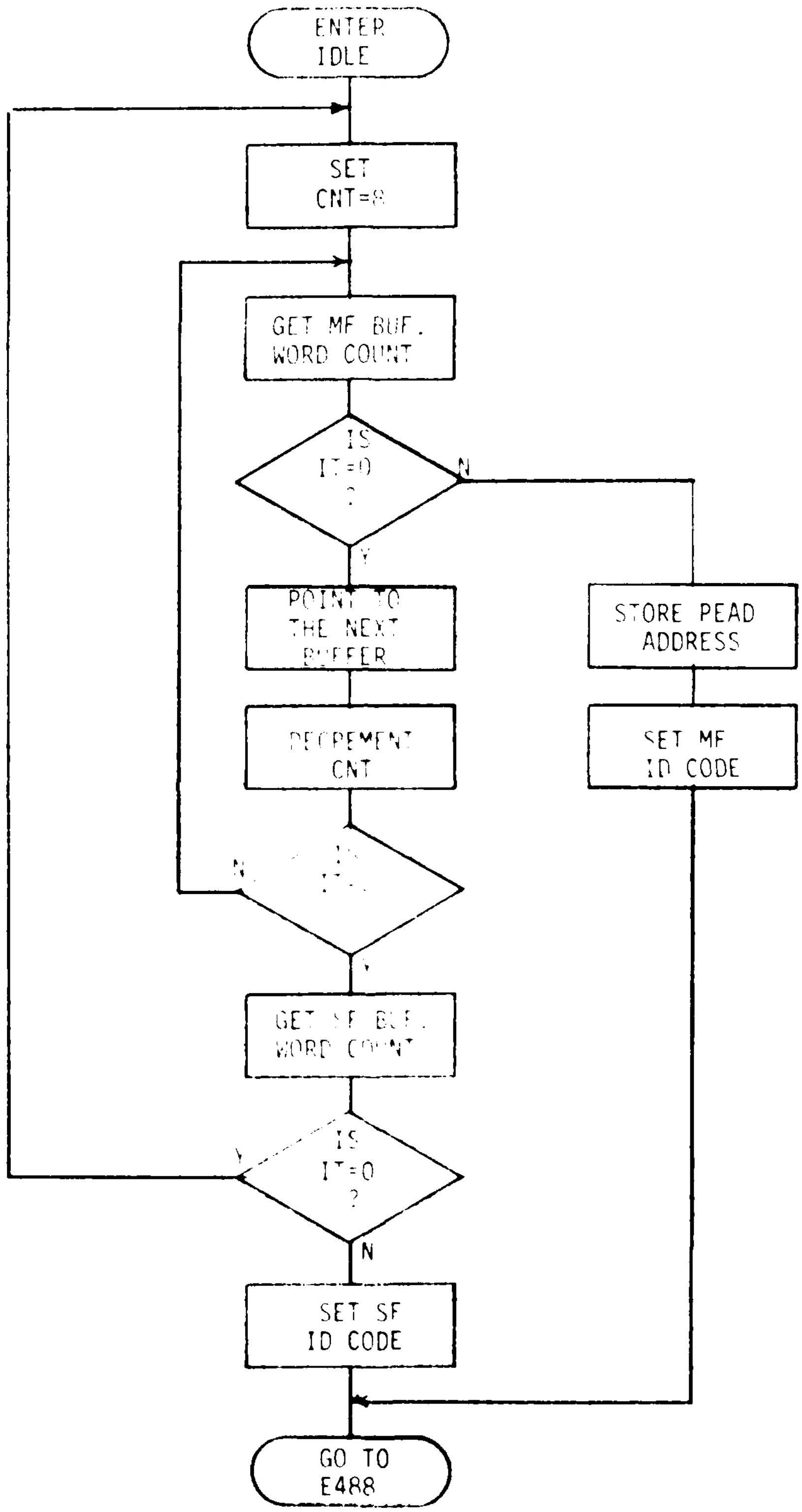
FS1 (CONT.)


ENTER FS2
OUTPUT SYNC PULSE
IS ERROR FLAG SET ?
Y
N
GET DATA
IS IT=90H ?
Y
N
RESET ERROR FLAG
RESET DCM PROGRAM COUNTER
GO TO BK
DISABLE INTERRUPT
ALL LEDS OFF
GET DATA
IS IT EBH ?
N
Y
RESET BIT COUNTER
ENABLE INTERRUPT
STORE DATA IN TEMP. STORAGE
GET DATA
Y
INTERRUPT ?
N
ENABLE 7.5 FF

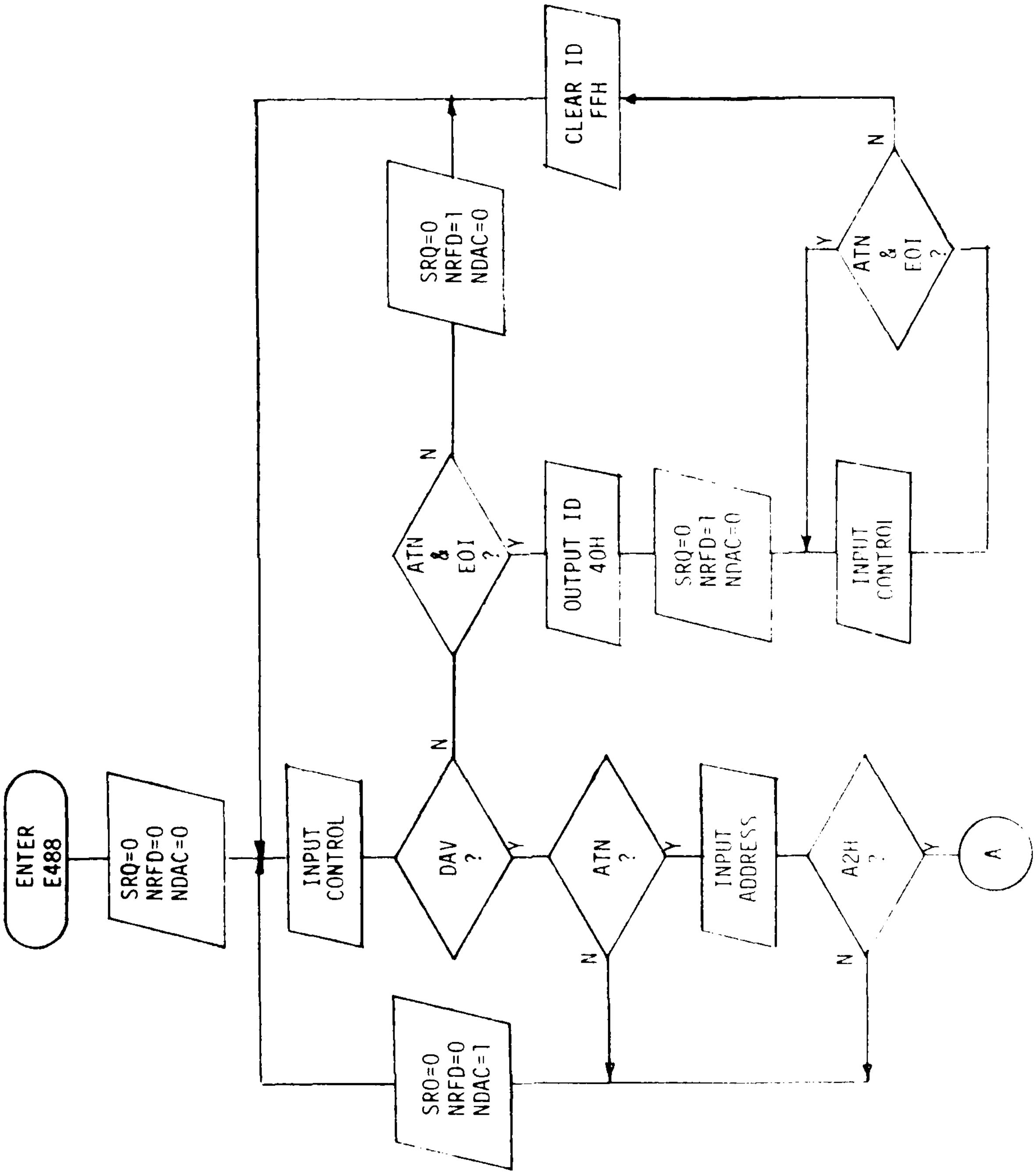
FS2


ENTER SEARCH
DISABLE INTERRUPT
GET DATA
IS IT EBH ?
N
Y
RESET BIT COUNTER
ENABLE 7.5 FF
INTERRUPT ?
N
Y
GET DATA
IS IT 90H ?
N
Y
RESET BIT COUNTER
DECREMENT LOOP CNTR 1
IS IT=0 ?
N
Y
INVERT DATA
INITIALIZE LOOP CNTR 1
IS IT=0 ?
N
Y
RESET LOOP CNTR 2
DECREMENT LOOP CNTR.2
ENABLE INTERRUPT
GO TO IDLE

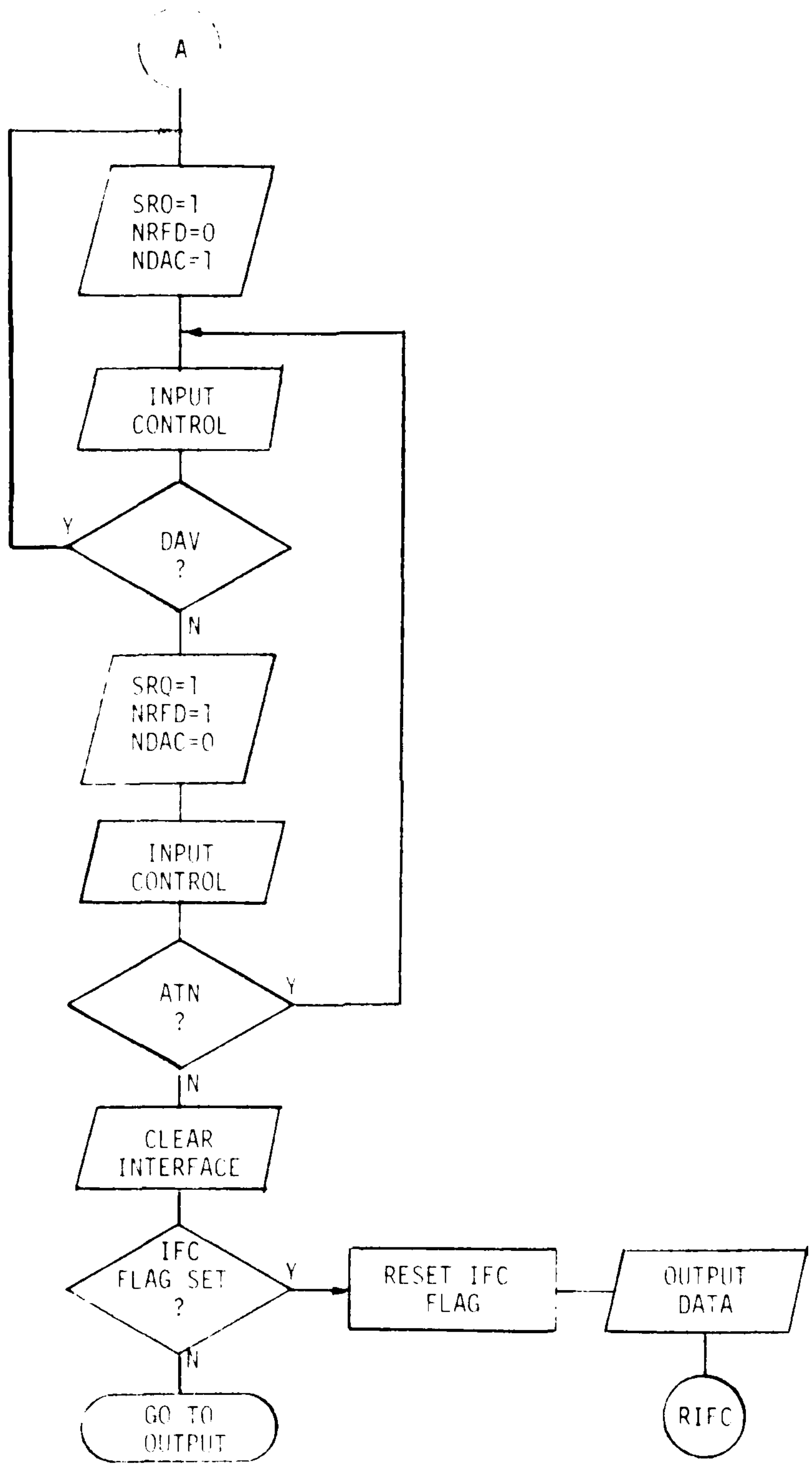
SEARCH


ENTER
IDLE
SET
GET MF BUF.
WORD COUNT
IS
?
N
Y
STORE READ
ADDRESS
DECREMENT
CNT
SET MF
ID CODE
N
Y
Y
IS
IT=0
?
N
SET SF
ID CODE
GO TO
E488

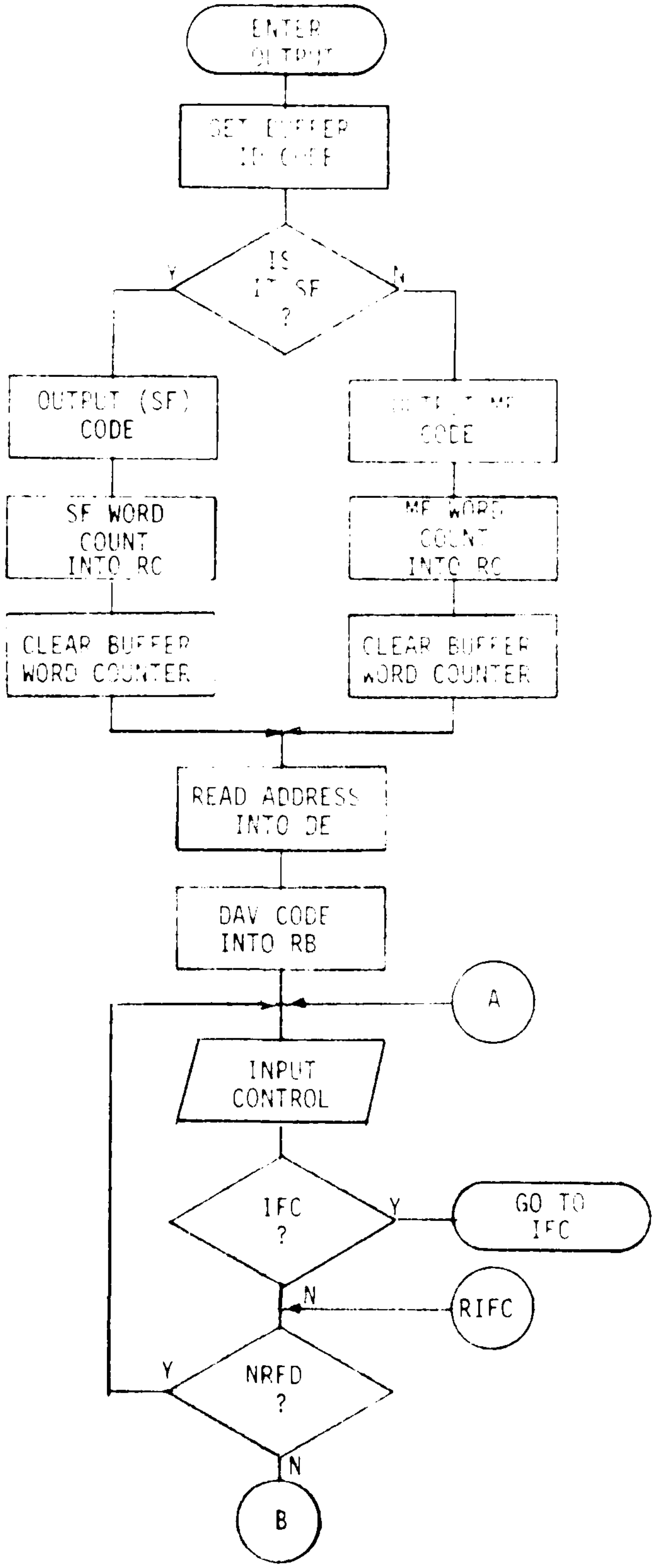
IDLE


ENTER E488
SRQ=0 NRFD=0 NDAC=0
INPUT CONTROL
DAV ?
ATN ?
INPUT ADDRESS
A2H ?
A
SRO=0 NRFD=0 NDAC=1
ATN & EOI ?
OUTPUT ID 40H
SRQ=0 NRFD=1 NDAC=0
INPUT CONTROL
SRQ=0 NRFD=1 NDAC=0
CLEAR ID FFH
ATN & EOI ?
Y
N

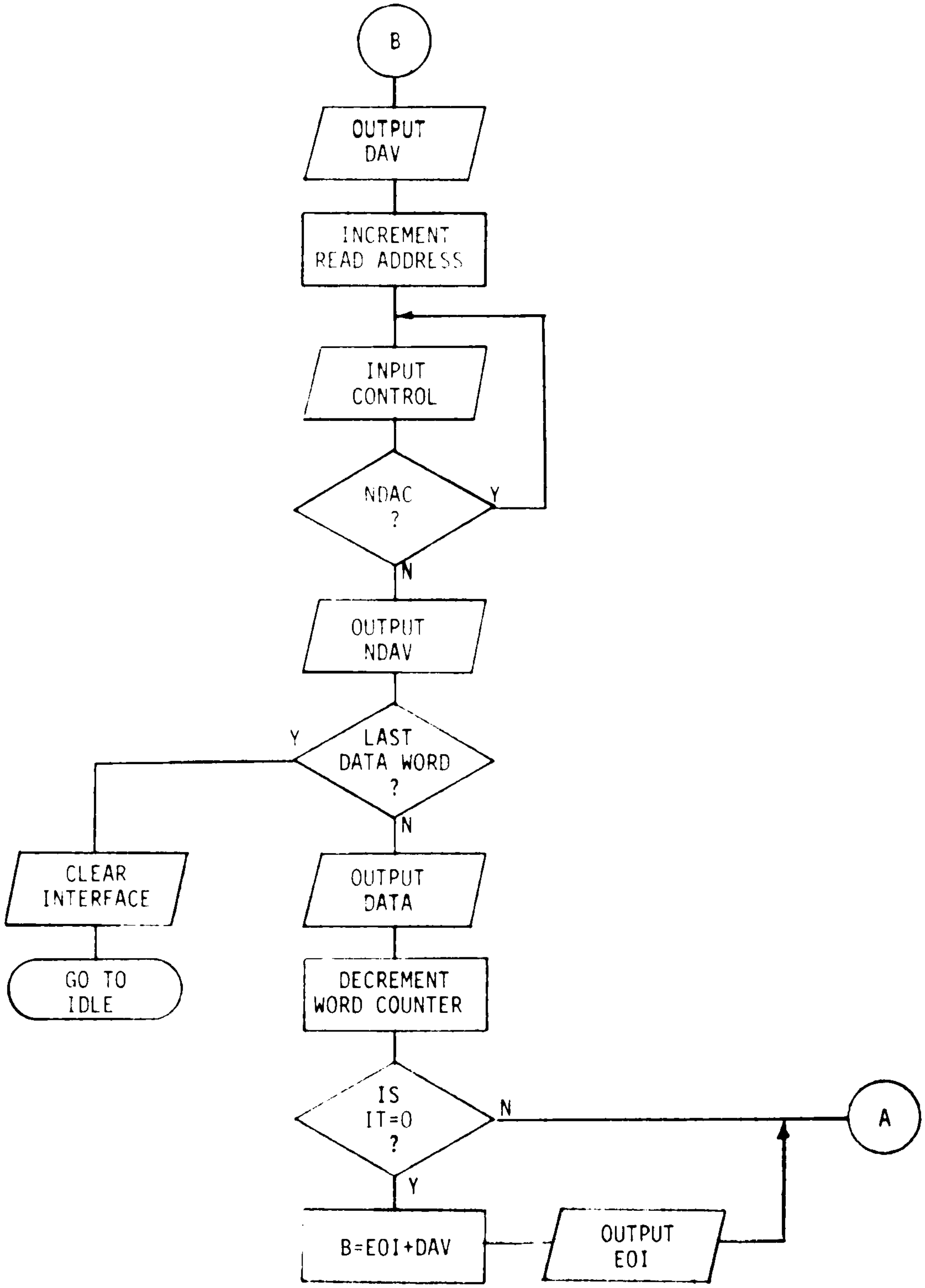
E488


A
SRQ=1
NRFD=0
NDAC=1
INPUT
CONTROL
Y
DAV
?
N
SRQ=1
NRFD=1
NDAC=0
INPUT
CONTROL
ATN
?
Y
N
CLEAR
INTERFACE
IFC
FLAG SET
?
Y
RESET IFC
FLAG
OUTPUT
DATA
N
GO TO
OUTPUT
RIFC

E488 CONT.


ENTER OUTPUT
SET BUFFER IN CHEF
IS IT SF ?
Y
N
OUTPUT (SF) CODE
SF WORD COUNT INTO RC
CLEAR BUFFER WORD COUNTER
MF WORD COUNT INTO RC
CLEAR BUFFER WORD COUNTER
READ ADDRESS INTO DE
DAV CODE INTO RB
A
INPUT CONTROL
IFC ?
Y
GO TO IFC
N
RIFC
NRFD ?
Y
N
B

OUTPUT



<u>OUTPUT CONT.</u>

```
[illegible]
[illegible] C3[illegible]   JMP   [illegible]
[illegible] C3[illegible]   JMP   WRI     ;[illegible]
[illegible] C3[illegible]   JMP   W[illegible]
[illegible] C3[illegible]   JMP   WRD     ;[illegible]
[illegible] C3[illegible]   JMP   TTY     ;[illegible]
[illegible] C3[illegible]   JMP   WR[illegible]
[illegible] C3[illegible]   JMP   W[illegible]
[illegible] C3[illegible]   JMP   [illegible]
[illegible] C3[illegible]   JMP   WRI     ;[illegible]
[illegible] C3[illegible]   JMP   EI
[illegible] C3[illegible]   JMP   RI
[illegible] C3[illegible]   JMP   F 1     ;[illegible]
[illegible] C3[illegible]   JMP   [illegible]
[illegible] [illegible]     ORG   44H
[illegible] [illegible] INIT  LXI   [illegible]
[illegible] [illegible]     MVI   [illegible]
[illegible] [illegible]     MVI   [illegible]
[illegible] 32[illegible]   STA   M[illegible]
[illegible] [illegible]     LXI   [illegible]
[illegible] 3E40        MVI   A,40H   ;TIMER [illegible]
[illegible] [illegible]     OUT   IOP4
[illegible] 3E[illegible]   MVI   A,[illegible]   ;TIMER [illegible]
[illegible] D314        OUT   IOP3
[illegible] 3E[illegible]   MVI   A,[illegible]   ;[illegible]
[illegible] D3[illegible]   OUT   IOP[illegible]
[illegible] 3E[illegible]   MVI   A,[illegible]   ;[illegible]
[illegible] D31B        OUT   IOPA    ;[illegible]
[illegible] 3E[illegible]   MVI   A,[illegible]   ;USART [illegible] BITS, [illegible]
[illegible] D331        OUT   IOPC    ;PARITY DISABLE
[illegible] 3E11        MVI   A,11H
[illegible] D331        OUT   IOPC
[illegible] 213[illegible]  LXI   H,003[illegible]H ;INITIAL [illegible] PROGRAM
[illegible] 22[illegible]10 SHLD  IMPA    ;POINTER
00EB 210009      LXI   H,0900H ;SUBFRAME DATA STORAGE ADDRE[illegible]
00EE 222210      SHLD  SBAL
00F1 222410      SHLD  SBPL    ;POINTER
00F4 222610      SHLD  RBAL    ;READ ADDRESS
00F7 210008      LXI   H,0800H ;MINOR FRAME STORAGE ADDRE[illegible]
00FA 222810      SHLD  SMAL
00FD 222A10      SHLD  SMPL    ;POINTER
0100 222C10      SHLD  RMAL    ;READ ADDRESS
0103 21000B      LXI   H,0B00H ;TTY STORAGE ADDRESS
0106 222E10      SHLD  TYSL
0109 223010      SHLD  TYRL    ;READ ADDRESS
010C AF          XRA   A       ;CLEAR MEMORY LOCATIONS
010D 320010      STA   M1
0110 320110      STA   M2
0113 320210      STA   M3
0116 320610      STA   WDC
0119 320710      STA   SFC
011C 320D10      STA   IFCF
011F 320008      STA   0800H
0122 322008      STA   0820H
```

```
0125 324008         STA    0840H
0128 326008         STA    0860H
012B 328008         STA    0880H
012E 32A008         STA    08A0H
0131 32C008         STA    08C0H
0134 32E008         STA    08E0H
0137 320009         STA    0900H
013A 32000A         STA    0A00H
013D D313           OUT    IOF2     ;CLEAR FUNCTION CONTROL
013F D319           OUT    IOF8     ;CLEAR IF DATA
0141 D31A           OUT    IOF9     ;CLEAR IB DATA
0143 3E1A           MVI    A,1AH    ;ENABLE 7.5 INTERRUPT
0145 30             SIM
0146 FB             EI
0147 C31D03         JMP    SEARCH
014A E5       XFR   PUSH   H        ;SAVE REGISTER CONTENTS
014B D5             PUSH   D
014C C5             PUSH   B
014D F5             PUSH   PSW
014E DB18           IN     IOF1     ;INPUT DATA
0150 320310         STA    M4
0153 2A2010         LHLD   DMFA     ;LOAD DCM PROGRAM ADDRESS
0156 110300         LXI    D,0003H  ;PUT 0003 INTO THE DE REGISTER PAIR
0159 19             DAD    D        ;ADD THE DE REGISTER PAIR TO HL = NEW DCM ADDR
015A 222010         SHLD   DMFA     ;STORE THE NEW DCM ADDRESS
015D E9             PCHL            ;GOTO TO DCM
015E F1       BK    POP    PSW      ;RESTORE REGISTERS
015F C1             POP    B
0160 D1             POP    D
0161 E1             POP    H
0162 FB             EI
0163 C9             RET             ;RETURN TO DATA PROCESSING
0164 3A0310   WRD1  LDA    M4       ;GET PCM WORD FROM TEMP. STORAGE
0167 07             RLC             ;IN THIS FRAME?
0168 D28001         JNC    W1
016B 210410         LXI    H,DVF    ;SET DV FLAG
016E 3680           MVI    M,80H
0170 210810         LXI    H,PAUSE  ;SET DV COUNTER
0173 3603           MVI    M,03H
0175 3A0110         LDA    M2       ;SET DV LED ON BIT
0178 F610           ORI    10H
017A 320110         STA    M2       ;STORE CONTROL WORD
017D C38501         JMP    W2
0180 210410   W1    LXI    H,DVF    ;RESET DV FLAG
0183 3600           MVI    M,00H
0185 3A0310   W2    LDA    M4       ;GET PCM WORD
0188 0F             RRC
0189 0F             RRC
018A 0F             RRC             ;TTY IN THIS FRAME?
018B D2A301         JNC    W3       ;IF NOT EXIT
018E 210510         LXI    H,TTF    ;SET TTY FLAG
0191 3680           MVI    M,80H
0193 210910         LXI    H,PTTY   ;SET TTY LED COUNTER
```

```
0196 3603             MVI   M,03H
0198 3A0110           LDA   M2      ;SET TTY ON BIT
019B F608             ORI   08H
019D 320110           STA   M2      ;STORE CONTROL WORD
01A0 C3A601           JMP   W4
01A3 3A0110   W3      LDA   M2
01A6 D313     W4      OUT   IOP2    ;OUTPUT LED CONTROL
01A8 3A0410   WPD     LDA   DVF     ;DV FLAG TO ACCUM
01AB FE80             CPI   80H     ;IS FLAG SET?
01AD C25E01           JNZ   BK      ;IF NOT,EXIT
01B0 3A0310           LDA   M4      ;LOAD THE WORD
01B3 2A0A10           LHLD  SMPL    ;POINT TO BUFFER
01B6 2C               INR   L       ;INCREMENT POINTER
01B7 77               MOV   M,A     ;STORE WORD
01B8 220A10           SHLD  SMPL    ;STORE POINTER
01BB 210610           LXI   H,WDL   ;POINT TO WORD COUNTER
01BE 34               INR   M       ;INCREMENT
01BF C35E01           JMP   BK      ;EXIT
01C2 3A0510   TTY     LDA   TTF     ;LOAD THE TTY FLAG
01C5 FE80             CPI   80H
01C7 C2D901           JNZ   T1      ;IF FLAG NOT SET JUMP
01CA 3A0310           LDA   M4      ;GET THE WORD
01CD 2A0E10           LHLD  TYSL    ;POINT TO BUFFER
01D0 77               MOV   M,A     ;STORE
01D1 2C               INR   L       ;INCREMENT POINTER
01D2 220E10           SHLD  TYSL    ;STORE POINTER
01D5 AF               XRA   A       ;CLEAR TTY FLAG
01D6 320510           STA   TTF
01D9 DB31     T1      IN    IOPC    ;USART STATUS CODE
01DB 0F               RRC           ;EMPTY?
01DC D25E01           JNC   BK      ;EXIT IF FULL
01DF 3A0E10           LDA   TYSL    ;LO BYTE OF STORE ADDRESS
01E2 2A3010           LHLD  TYPT    ;READ ADDRESS
01E5 BD               CMP   L       ;STR ADRS = READ ADRS?
01E6 C2F001           JNZ   T2      ;IF NOT EMPTY, JUMP
01E9 3E11             MVI   A,11H   ;END OF X-MISSION CODE
01EB D331             OUT   IOPC    ;TO USART
01ED C35E01           JMP   BK      ;EXIT
01F0 7E       T2      MOV   A,M     ;DATA TO USART
01F1 D330             OUT   IOPB
01F3 3E31             MVI   A,31H
01F5 D331             OUT   IOPC
01F7 2C               INR   L       ;INCREMENT READ ADDRESS
01F8 223010           SHLD  TYPT    ;STORE
01FB C35E01           JMP   BK      ;EXIT
01FE 3A0110   OUTPUT  LDA   M2
0201 F604             ORI   04H
0203 320110           STA   M2
0206 D313             OUT   IOP2
0208 210A10           LXI   H,PTAP
020B 3603             MVI   M,03H
020D 3A0C10           LDA   DATID
0210 FE06             CPI   06H     ;IS IT SF?
```

```
0212 C22002          JNZ     MF       ;IF NOT, GO TO MF
0215 D319            OUT     IOP8     ;OUTPUT CODE
0217 2A2610          LHLD    RBAL     ;SF READ ADDRESS
021A 4E              MOV     C,M      ;WORD CNTR
021B 3600            MVI     M,0H     ;CLEAR FULL FLAG
021D C32802          JMP     AU
0220 D319     MF     OUT     IOP8     ;OUTPUT MF CODE
0222 2A2C10          LHLD    RMAL     ;MF READ ADDRESS
0225 4E              MOV     C,M      ;WORD CNTR
0226 3600            MVI     M,0H     ;CLEAR FULL FLAG
0228 EB       AU     XCHG
0229 0620            MVI     B,20H    ;DAV INTO B
022B DB12     AUT    IN      IOP6     ;INPUT 488 CONTROL WORD
022D E602            ANI     02H      ;IFC?
022F CA6602          JZ      IFC
0232 DB12     RIFC   IN      IOP6
0234 0F              RRC              ;NRFDHI
0235 D22B02          JNC     AUT
0238 07              RLC
0239 07              RLC
023A DA2B02          JC      AUT
023D 78              MOV     A,B
023E D31A            OUT     IOP9     ;OUTPUT DAV
0240 1C              INR     E        ;INCREMENT READ POINTER
0241 DB12     A1     IN      IOP6     ;INPUT CONTROL
0243 07              RLC              ;NDAC HI?
0244 D24102          JNC     A1
0247 AF              XRA     A
0248 D31A            OUT     IOP9     ;RESET DAV
024A B1              ORA     C        ;IS THIS THE LAST WORD
024B CA5E02          JZ      CLR      ;IF YES,EXIT
024E 1A              LDAX    D        ;DATA INTO ACCUM.
024F D319            OUT     IOP8
0251 0D              DCR     C        ;DECREMENT WORD COUNTER
0252 C22B02          JNZ     AUT      ;MORE DATA?
0255 0624            MVI     B,24H    ;PUT EOI.DAV INTO B
0257 3E04            MVI     A,04H    ;OUTPUT EOI
0259 D31A            OUT     IOP9
025B C32B02          JMP     AUT
025E AF       CLR    XRA     A
025F D319            OUT     IOP8     ;CLEAR INTERFACE
0261 D31A            OUT     IOP9
0263 C37102          JMP     IDLE     ;EXIT
0266 AF       IFC    XRA     A        ;CLEAR I STA INTERFACE
0267 D319            OUT     IOP8
0269 3E80            MVI     A,80H    ;SET IFC FLAG
026B 320D10          STA     IFCF
026E C39B02          JMP     E4RE
0271 0E08     IDLE   MVI     C,8H     ;ESTABLISH COUNTER
0273 2A2C10          LHLD    RMAL     ;POINT TO MF BUFFER
0276 AF       I1     XRA     A        ;CHECK WORD COUNT
0277 B6              ORA     M
0278 C29302          JNZ     I2       ;IF ZERO INCREMENT POINTER
```

```
027B 7D              MOV   A,L
027C C620            ADI   20H
027E 6F              MOV   L,A
027F 0D              DCR   C        ;DECREMENT COUNTER
0280 C27602          JNZ   I1       ;IF NOT ZERO TRY AGAIN
0283 2A2610          LHLD  RBAL     ;POINT TO SE BUFFER
0286 AF              XRA   A
0287 B6              ORA   M
0288 CA7102          JZ    IDLE     ;IF WORD COUNT ZERO, RETURN TO IDLE
028B 210C10          LXI   H,DATID
028E 3606            MVI   M,06H    ;SET SEID CODE
0290 C39B02          JMP   E488
0293 222C10   I2     SHLD  RMAL     ;STORE READ ADDRESS
0296 210C10          LXI   H,DATID
0299 3607            MVI   M,07H    ;SET MEID CODE
029B 3A0110   E488   LDA   M2
029E F602            ORI   02H
02A0 320110          STA   M2
02A3 D313            OUT   IOF2
02A5 210B10          LXI   H,F
02A8 3603            MVI   M,03H
02AA 3E08            MVI   A,08H
02AC D31A            OUT   IOF9     ;CLEAR INTERFACE
02AE DB12     DAV    IN    IOF6     ;INPUT CONTROL WORD
02B0 E620            ANI   20H      ;DAV?
02B2 CAC302          JZ    AIN      ;YES, GO TO AIN
02B5 DB12            IN    IOF6
02B7 E614            ANI   14H      ;AIN & EOI?
02B9 CA0803          JZ    ATE      ;GO ON IF YES
02BC 3E88            MVI   A,88H    ;RFDSRQ
02BE D31A            OUT   IOF9     ;OUTPUT RFD
02C0 C3AE02          JMP   DAV      ;TRY AGAIN
02C3 DB12     AIN    IN    IOF6     ;INPUT
02C5 E610            ANI   10H      ;ATN?
02C7 CAD102          JZ    MAD      ;TRY AGAIN
02CA 3E09     DA     MVI   A,09H    ;DAL
02CC D31A            OUT   IOF9
02CE C3AE02          JMP   DAV      ;TRY AGAIN
02D1 DB11     MAD    IN    IOF5     ;INPUT ADDRESS
02D3 F680            ORI   80H      ;SET BIT 8=1
02D5 FEA2            CPI   0A2H
02D7 C2CA02          JNZ   DA       ;IF NO, TRY AGAIN
02DA 3E01     MD     MVI   A,01H    ;NRFD
02DC D31A            OUT   IOF9     ;OUTPUT RFD
02DE DB12     MD1    IN    IOF6
02E0 E620            ANI   20H
02E2 CADA02          JZ    MD
02E5 3E80            MVI   A,80H
02E7 D31A            OUT   IOF9
02E9 DB12            IN    IOF6
02EB E610            ANI   10H      ;ATN?
02ED CADE02          JZ    MD1      ;TRY AGAIN
02F0 AF              XRA   A
```

```
[illegible] ACM V3.3 [illegible]

[illegible] 02E1 [illegible]21A              [illegible]     IOP[illegible]
[illegible] 02E[illegible] [illegible]          LDA     [illegible]
[illegible] 02EA FE[illegible]               [illegible]     [illegible]
00?23 02E8 CAEE02                 J[illegible]      E[illegible]
[illegible] 02EB C3EE01                 JMP     [illegible]
[illegible] 02EE AF        E[illegible]          XRA     A
[illegible] 02EF 3[illegible]               STA     [illegible]
[illegible] 030[illegible] 1A                 [illegible]     [illegible]
[illegible] 030[illegible] D31[illegible]                 OUT     [illegible]
00329 0305 C33[illegible]               JMP     [illegible]
00330 0308 3E40          ACE     MVI     A,40H     ;[illegible]
00331 0[illegible] D31[illegible]                 OUT     IOP[illegible]
00332 030C 3E[illegible]                  MVI     A,[illegible]H    ;[illegible]
00333 030E D31A                   [illegible]     IOP[illegible]     ;[illegible]
00334 0310 DB1[illegible]         AE      IN      IOP[illegible]
00335 0312 E614                   ANI     14H       ;[illegible]
00336 0314 CA10[illegible]                 J[illegible]      AE
0033[illegible] 031[illegible] AF                     XRA     A
00338 031[illegible] D31[illegible]                   OUT     IOP[illegible]
00339 031A C3[illegible]                  J[illegible]      [illegible]
00340 031D FB            [illegible]   [illegible]                 ;[illegible] INTERRUPT
00341 031E DB18          S1      IN      IOP1      ;READ IN W[illegible]
00342 0320 FEEB                   CPI     0EBH      ;[illegible] WORD [illegible]
00343 0322 CA[illegible]                 JZ      S4        ;BRANCH IF [illegible]
00344 0325 0B            S3      DCX     B         ;DECREMENT [illegible] COUNTER
00345 0326 78                     MOV     A,B       ;[illegible] COUNTER [illegible]
00346 0327 B[illegible]                     ORA     A
003[illegible] 0328 C21E03                 JNZ     S1        ;IF NOT, BRANCH
00348 032B 79                     MOV     A,C
00349 032C B7                     ORA     A
00350 032D C21E03                 JNZ     S1
00351 0330 3A0110                 LDA     M[illegible]        ;[illegible]
00352 0333 EE20                   XRI     20H
00353 0335 320110                 STA     M2
00354 0338 D313                   OUT     IOP2
00355 033A 0127[illegible]        S[illegible]      LXI     B,[illegible]H ;RESET LOOP COUNTER
00356 033D C31E03                 JMP     S1        ;JUMP BACK TO START
00357 0340 210110        S4      [illegible]     H,M2      ;SET [illegible]
00358 0343 3E01                   MVI     A,01H
00359 0345 B6                     ORA     M
00360 0346 320110                 STA     M2
00361 0349 D313                   OUT     IOP2
00362 034B 3EFE                   MVI     A,0FEH
00363 034D A6                     ANA     M
00364 034E 320110                 STA     M2
00365 0351 D313                   OUT     IOP2
00366 0353 3E1C                   MVI     A,1CH
00367 0355 30                     SIM                ;SET INT 7.5 MASK
00368 0356 FB                     EI                 ;ENABLE INTERUPT
00369 0357 20            S5      RIM                ;WAIT UNTIL 7.5 IS ENABLED
00370 0358 E640                   ANI     40H
00371 035A CA5703                 JZ      S5
00372 035D DB18                   IN      IOP1      ;READ IN NEW WORD
```

```
00373 035F FE90               CPI    90H      ;IS WORD 90?
00374 0361 C22503             JNZ    S3
00375 0364 210110             LXI    H,M2     ;SET CLOCK
00376 0367 3E01               MVI    A,01H
00377 0369 B6                 ORA    M
00378 036A 320110             STA    M2
00379 036D D313               OUT    IOP2
00380 036F 3EFE               MVI    A,0FEH
00381 0371 A6                 ANA    M
00382 0372 320110             STA    M2
00383 0375 D313               OUT    IOP2
00384 0377 3A3210             LDA    M5       ;DECREMENT LOOP COUNTER
00385 037A 3D                 DCR    A
00386 037B 323210             STA    M5
00387 037E C23A03             JNZ    S5       ;IF NOT ZERO JUMP
00388 0381 3E02               MVI    A,02H    ;RESET LOOP COUNTER
00389 0383 323210             STA    M5
00390 0386 3E19               MVI    A,19H    ;UNMASK INTERUPT
00391 0388 30                 SIM
00392 0389 FB                 EI
00393 038A C37102             JMP    IDLE     ;GOTO IDLE
00394 038D 3A0310      SFID   LDA    M4       ;GET SFID FROM TEMP STORAGE
00395 0390 B7                 ORA    A        ;IS SFID=0 ?
00396 0391 C2BA03             JNZ    SF1      ;BRANCH TO END IF NOT
00397 0394 [illegible]        LHLD   SBPL     ;STORE SF WORD COUNTER IN FIRST LOCATION OF BUFFER
00398 0397 2E00               MVI    L,00H
00399 0399 3A0[illegible]10   LDA    SFC
00400 039C [illegible]        MOV    M,A
00401 039D AF                 XRA    A        ;SET SF WORD COUNTER TO 0
00402 039E [illegible]        STA    SFC
00403 03A1 3E09               MVI    A,09H    ;IS STORE ADDR. 09XXH?
00404 03A3 BC                 CMP    H
00405 03A4 [illegible]        JNZ    SF2      ;BRANCH IF IT IS NOT
00406                                         ;SET STORE ADDR AND POINTER TO 0A00H
00407                                         ;SET READ TO 0900H
00408 03A7 3E0A               MVI    A,0AH    ;ELSE SET STORE ADDR AND POINTER TO 0900H
00409                                         ;SET READ TO 0A00H
00410 03A9 [illegible]  SF[illegible]  STA    SBPH
00411 03AC [illegible]        STA    SBAH
00412 03AF AF                 XRA    A
00413 03B0 [illegible]        STA    SBPL
00414 03B3 [illegible]        SHLD   RBAL
00415 03B6 [illegible]  SF1   JMP    BE       ;EXIT
00416 03B9 3A0310      SFWORD LDA    M3       ;GET DATA FROM TEMP STORAGE
00417 03BC [illegible]        LHLD   SBPL     ;INCREMENT STORAGE BUFFER
00418 03BF 2C                 INR    L
00419 03C0 [illegible]        SHLD   SBPL
00420 03C3 77                 MOV    M,A      ;STORE DATA
00421 03C4 [illegible]        LDA    SFC      ;INCREMENT SUBFRAME WORD COUNTER
00422 03C7 3C                 INR    A
00423 03C8 [illegible]        STA    SFC
00424 03CB C35E01             JMP    BE       ;EXIT
00425 03CE 3A0[illegible]  [illegible]  LDA  M3  ;BRING IN DATA
```

[illegible] A M [illegible]

```
[illegible]                          CPI     [illegible]   ;[illegible]
[illegible]                          [illegible] [illegible] ;BRANCH IF [illegible]
[illegible]                          LHLD    SMAL
[illegible]                          MVI     A,[illegible]H ;SET [illegible] FLAG
[illegible]                          STA     M[illegible]
[illegible]                          INX     H       ;SET [illegible] [illegible] WORD
[illegible]                          MVI     A,[illegible]
[illegible]                          ORA     A
[illegible]                          MOV     M,A
[illegible]                          DCR     E
[illegible]                          JMP     [illegible]
[illegible]             [illegible]  LDA     DV[illegible] ;[illegible] DATA VALID FLAG [illegible]
[illegible]                          ORA     A
[illegible]                          JZ      [illegible] ;IF [illegible] BRANCH TO [illegible]
00440 03EE [illegible]               LHLD    SMAL    ;[illegible] STORE [illegible]
00441 03F1 [illegible]  [illegible]  SHLD    SMEL    ;STORE [illegible] [illegible]
00442 03F4 [illegible]               LDA     WHO     ;STORE W[illegible] IN [illegible]
00443 03F7 [illegible]               MOV     M,A
00444 03F8 AF                        XRA     A       ;CLEAR WHO
00445 03F9 [illegible]               STA     WHO
00446 03FC 3E20                      MVI     A,20H   ;ADD 20H TO [illegible] POINTER AND [illegible]
00447 03FE 85                        ADD     L
00448 03FF 32[illegible]10           STA     SMEL
00449 0402 32[illegible]10           STA     SMAL
00450 0405 6F                        MOV     L,A
00451 0406 AF                        XRA     A       ;CLEAR STORE BUFFER
00452 0407 77                        MOV     M,A
00453 0408 3A[illegible]10           LDA     RMAL    ;IS READ [illegible] [illegible] STORE [illegible]
00454 040B BD                        CMP     L
00455 040C C2[illegible]04           JNZ     B1      ;BRANCH IF NO
00456 040F C620                      ADI     20H     ;ADD 20H TO READ ADDR
00457 0411 32[illegible]10           STA     RMAL
00458 0414 3A0910       B?           LDA     FITY    ;IS ITY COUNTER 0?
00459 041? FE00                      CPI     00H
00460 0419 C2[illegible]04           JNZ     B1      ;BRANCH IF NOT
00461 041C 3A0110                    LDA     M2      ;RESET ITY LED ON BIT
00462 041F E6F[illegible]            ANI     0F[illegible]H
00463 0421 320110                    STA     M2
00464 0424 C3[illegible]04           JMP     B2
00465 042? 3D           B1           DCR     A       ;DECREMENT COUNTER
00466 0428 320910                    STA     FITY
00467 042B 3A0A10       B2           LDA     FTAK    ;IS TALK COUNTER 0?
00468 042E FE00                      CPI     00H
00469 0430 C23E04                    JNZ     B3      ;BRANCH IF NOT
00470 0433 3A0110                    LDA     M2      ;RESET TALK LED ON BIT
00471 0436 E6FB                      ANI     0FBH
00472 0438 320110                    STA     M2
00473 043B C34204                    JMP     B4
00474 043E 3D           B3           DCR     A       ;DECREMENT COUNTER
00475 043F 320A10                    STA     FTAK
00476 0442 3A0B10       B4           LDA     F?      ;IS COUNTER 0?
00477 0445 FE00                      CPI     00H
00478 0447 C25504                    JNZ     B5      ;BRANCH IF NOT
```

```
[illegible] 044A 3A0110           LDA   M2       ;RE[illegible] COUNTER [illegible]
[illegible] 044D E6FE             ANI   0FDH
[illegible] 044F 320110           STA   M2
[illegible] 0452 C35904           JMP   B6
[illegible] 0455 3D         B5    DCR   A        ;DECREMENT COUNTER
[illegible] 0456 320B10           STA   F
[illegible] 0459 3A0010     B6    LDA   FAUSE    ;[illegible] COUNTER [illegible]
[illegible] FE0[illegible]        CPI   0?H
[illegible] C26C04                JNZ   B        ;[illegible] BRANCH
[illegible] 3A0110                LDA   M2
[illegible] E6FE                  ANI   0FEH
[illegible]                       STA   M2
[illegible]                       JMP   E1
[illegible] 3D              B     DCR   A        ;[illegible] COUNTER
[illegible]                       STA   FAU[illegible]
[illegible]                 E[illegible]  LDA   M2
[illegible] D313                  OUT   13H      ;[illegible] CONTROL
[illegible]                       JMP   E1       ;EXIT
[illegible]                       LXI   H,M      ;[illegible]
[illegible]                       MVI   A,[illegible]H
[illegible]                       ORA   M
[illegible] D313                  OUT   13H
[illegible]                       MVI   A,[illegible]H
[illegible] A6                    ANA   M
[illegible] D313                  OUT   13H
[illegible]                       LDA   M3       ;[illegible]
[illegible]                       ORA   A        ;[illegible]
[illegible]                       JNZ   E1
[illegible]                 E1    LDA   M3       ;[illegible] WITH [illegible]
[illegible]                       CPI   90H      ;[illegible]
[illegible]                       JNZ   E1       ;[illegible]
[illegible]                       XRA   A        ;[illegible]
[illegible]                       STA   M3
[illegible]                       LXI   H,DM12   ;[illegible]
[illegible]                       SHLD  [illegible]A
[illegible]                       JMP   EE       ;[illegible]
[illegible]                 E1    EI             ;[illegible]
[illegible]                       LDA   M        ;[illegible]
[illegible]                       ANI   0F1H
[illegible]                       STA   M2
[illegible] D3[illegible]         OUT   10H2
[illegible]                 F     IN    10H1     ;READ [illegible] WORD
[illegible] FEEB                  CPI   0EBH     ;[illegible] WORD [illegible]
[illegible]                       JNZ   F[illegible]  ;BRANCH IF [illegible]
[illegible]                       LXI   H,M[illegible]  ;[illegible]
[illegible] 3E01                  MVI   A,01H
[illegible] B6                    ORA   M
[illegible]                       STA   M2
[illegible] D3[illegible]         OUT   10H[illegible]
[illegible] 3EFE                  MVI   A,0FEH
[illegible] A6                    ANA   M
[illegible]                       STA   M2
[illegible] D313                  OUT   10H[illegible]
```

```
[illegible]                                                   [illegible]

[illegible]                    MVI     [illegible]
[illegible]                    SIM             [illegible]
[illegible]                    EI              [illegible]
[illegible]                    SIM             [illegible]
[illegible]                    CPI     [illegible]
[illegible]                    JZ      [illegible]
[illegible]                    IN      [illegible]     [illegible] READ IN NEW WORD
[illegible]                    JC      [illegible]
[illegible]                    MVI     [illegible]     [illegible]
[illegible]                    SIM
[illegible]                    EI
[illegible]                    JMP     [illegible]     [illegible]
[illegible]                    END     [illegible]
```

IV. REFERENCES

1. [illegible], R. and Rochefort, J.S. "Control and Data Transmission System for a Balloon-Borne Ion Mass Spectrometer", Proceedings International Telemetering Conference, October [illegible], Vol. [illegible], pp. 33[illegible]-341.

2. Sukys, R. and Rochefort, J.S. "Control and Data Transmission System for a Balloon-Borne Ion Mass Spectrometer", Scientific Report No. 3, Contract No. F1962[illegible], October 198[illegible], AFGL-TR-81-0202, ADA104512.

3. J. Spencer Rochefort, Raimundas Sukys, [illegible] Air-Borne Quadrupole Ion Mass Spectrometer", [illegible] Contract No. F1962[illegible]-78-C-[illegible], [illegible], ADA115399.

4. IEEE STANDARD 4[illegible]-197[illegible] [illegible] Instrumentation", The IEEE, Inc., [illegible], New York, N.Y., November 1978.

5. EIA STANDARD RS-232-C "Interface Between [illegible] and Data Communications Equipment [illegible] Interchange", Electronic Industries Association, [illegible] Department, 2001 Eye St., N.W., Washington, [illegible].

V. PERSONNEL

A list of the engineers who contributed [illegible] given below:

J. Spencer Rochefort, Professor of Electrical Engineering [illegible] Principal Investigator.

Raimundas Sukys, Senior Research Associate, [illegible]

VI. RELATED CONTRACTS AND PUBLICATIONS

F19628-74-C-0042	1 September 1973 through October 1976
F19628-76-C-0256	1 August 1976 through 31 October 1978
F19628-78-C-0218	15 September 1978 through 14 September 1981
F19628-81-C-0162	15 September 1981 through present.

Raimundas Sukys, Steven Goldberg, "Control Circuits for Rocket Payload Neutralization Experiment and Other Topics", Scientific Report No. 1 for Contract F19628-74-C-0042, October 1974, AFCRL-TR-74-0580, ADA00[illegible].

R. Sukys, J. Spencer Rochefort, S. Goldberg, "Bias and Signal Processing Circuits for a Mass Spectrometer in the Project EXCEDE: SWIR Experiment", Scientific Report No. 2 for Contract F19628-74-C-0042, October 1975, AFGL-TR-76-0060, ADA026514.

J. Spencer Rochefort, Raimundas Sukys, "Instrumentation Systems for Mass Spectrometers", Final Report for Contract F19628-74-C-0042, September 1976, AFGL-TR-76-0200, ADA032313.

J. Spencer Rochefort, Raimundas Sukys, "A Digital Control Unit for a Rocket Borne Quadrupole Mass Spectrometer", Scientific Report No. 1, for Contract F19628-76-C-0256, April 1978, AFGRL-TR-78-0106, ADA057[illegible].

J. Spencer Rochefort, Raimundas Sukys, "Electronics for a Rocket Borne Quadrupole Cluster Ion Mass Filter", Final Report for Contract F19628-76-C-0256, October 1978, AFGL-TR-78-0292, ADA066289.

Gerousis, V.S., "A Programmable Control Unit for a Balloon-Borne Mass Spectrometer Based on Intel 8085A Microprocessor", Scientific Report No. 1, Contract AF19628-78-C-0218, September 1979, AFGL-TR-79-0225, ADA106398.

Palasek, T., "An RF Oscillator for Rocket-Borne and Balloon-Borne Mass Spectrometers", Scientific Report No. 2, Contract AF19628-78-C-0218, September 1979, AFGL-TR-79-0226, ADA078797.

Sukys, R. and Rochefort, J.S. "Control and Data Transmission System for a Balloon-Borne Ion Mass Spectrometer", Proceedings International Telemetering Conference, October 1980, Vol. XVI, pp. 335-341.

Sukys, R. and Rochefort, J.S. "Control and Data Transmission System for a Balloon-Borne Ion Mass Spectrometer", Scientific Report No. 3. Contract AF19628-78-C-0218, October 1980, AFGL-TR-81-0202, ADA104512.

J. Spencer Rochefort, Raimundas Sukys, "Control Electronics for Air-Borne Quadrupole Ion Mass Spectrometer", Final Report for Contract F19628-78-C-0218, October 1981, AFGL-TR-82-0056, ADA115399.

DA

FIL

9-